

```

/*****
** GRAPHIC SECTION **
*****/

////////////////////////////////////
// VARIABLES //////////////////////////////////////
////////////////////////////////////

// OBJECTS //////////////////////////////////////

class Spook { // disegna spook fluttuante, secondo l'umore scelto (felice, triste, paura,
esulta, racconta)
    Spook (String mood, int frameAttuale, boolean fumo) {

        if(frameAttuale%40 < 20) i=i+1;
        else i=i-1;

        // disegno o non disegno i fumini dietro?
        if(fumo==true) image(fumini,0,0);

        if(mood=="felice") {
            image(spook_felice, 130, 180+i/2);
        }
        else if(mood=="triste") {
            image(spook_triste, 130, 180+i/2);
        }
        else if(mood=="esulta") {
            image(spook_esulta, 130, 180+i/2);
        }
        else if(mood=="saluta") {
            image(spook_saluta, 130, 180+i/2);
        }
        else if(mood=="racconta") {
            image(spook_racconta, 130, 180+i/2);
        }
        else if(mood=="paura") {
            if(frameAttuale%3==0) k=+1;
            if(frameAttuale%3==1) k=-1;
            if(frameAttuale%3==2) k=0;
            image(spook_paura, 130+k, 180+i/4);
        }
        else if(mood=="giravolta") {

            image(spook_giravolta,k*86,0,86,92,130,180+i/2);
            ++j;
            k=j;
            if(j>8 && j<= 18) k = -9+j;
            if(j>6 && j<=8) k = 6;
            if(j>=16 && j<=18) k = 6;
            if(j>18) k=1; // da qui disegna spook fermo!
        }
        else if(mood=="occhei") {
            image(spook_occhei,k*83,0,83,92,130,180+i/2);
            if(k<4) {
                ++k;
            }
        }
        else if(mood=="piange") {
            image(spook_piange,k*75,0,75,92,130,180);
            if(frameAttuale%4==0) {
                if(k==0) j=+1;
                else if(k==4) j=-1;
                k = k+j;
            }
        }
    }
}

// PHOTO SLIDE

```

```

class emptyImage { // disegna spook fluttuante, secondo l'umore scelto (felice, triste, paura,
esulta, racconta)
    emptyImage (int imgX, int imgY) {
        fill(dark_yellow);
        strokeWeight(1);
        stroke(orange);
        rect(imgX, imgY, 36, 27);
        textAlign(CENTER);
        textFont(din_black_15);
        text("?", imgX+37/2, imgY+20);
        textFont(din_light_15_brown);
        textAlign(LEFT);
    }
}

// UN CORIANDOLO
void pallino(int pX, int pY, int pWidth, int pTransp, int pTime, int pColor) {
    noStroke();
    fill(255,0+pColor,0);
    ellipse(pX,pY,pWidth,pWidth);
}

// UN FIOCCO DI NEVE
void fiocco(int pX, int pY, int pWidth, int pTransp, int pTime, int pColor) {
    noStroke();
    fill(230,230,230);
    ellipse(pX,pY,pWidth,pWidth);
}

// PIOGGIA DI CORIANDOLI
void coriandoli(int numCoriandoli, String direzione) {
    // CORIANDOLI
    for(p=0; p<=numCoriandoli-2; p++) {
        println("PALLINO"+pallinoX[p]);
        pallinoTime[p] -= 1;

        // modifico la X e la Y per farlo muovere :o
        if(pallinoTime[p]<=0) {
            //if(frameCount%2==0)
            pallinoDirection[p] = random(-5,5);
            if(frameCount%2==0) pallinoX[p] = pallinoX[p]+pallinoDirection[p];
            pallinoX[p] = constrain(pallinoX[p],0+pallinoW[p]+3,width);
            //if(direzione=="salita")
            pallinoY[p] -= pallinoSpeed[p]; //random(4,10);
            //else pallinoY[p] = pallinoSpeed[p];
            pallinoColor[p] += 4;
        }

        // disegno il pallino solo se ? ancora nello schermo
        if(pallinoY[p]>-10){
            pallino(pallinoX[p],pallinoY[p],pallinoW[p],pallinoT[p],pallinoTime[p],pallinoColor[p]);
            pallino(pallinoX[p],pallinoY[p],pallinoW[p],pallinoT[p],pallinoTime[p],pallinoColor[p]);
        }

        // pioggia infinita di coriandoli
        if(numCoriandoli==100) {
            if(pallinoY[p]<=-10) {
                pallinoY[p] = random(330, 350);
                pallinoColor[p] = 150;
            }
        }
    }
}

// CADE LA NEVE
void neve(int numNeve) {
    for(int p=0; p<=25; p++) {
        println("ARRAY NEVEY "+p+" = "+neveY[p]);
        neveTime[p] -= 1;
    }
}

```

```

// modifico la X e la Y per farlo muovere :o
if(neveTime[p]<=0) {
    neveDirection[p] = random(-1,1);
    if(frameCount%4==0) neveX[p] = neveX[p]+neveDirection[p];
    neveX[p] = constrain(neveX[p],0+neveW[p]+3,width);
    neveY[p] += neveSpeed[p];
}

// disegno il pallino solo se ? ancora nello schermo
if(neveY[p]<340){
    fiocco(neveX[p],neveY[p],neveW[p],neveT[p],neveTime[p],neveColor[p]);
    fiocco(neveX[p],neveY[p],neveW[p],neveT[p],neveTime[p],neveColor[p]);
}

// pioggia infinita di coriandoli
if(neveY[p]>=340) {
    neveY[p] = random(-30, -50);
}
}

// RESETTA CORIANDOLI
void reset_coriandoli() {
    for(int p=0;p<=pallinoX.length-2;p++) {
        //create random coords
        int randomX = random(0, width);
        int randomY = random(330, 350);
        int randomW = random(1,8);
        int randomT = random(20,255);

        // store data
        pallinoX[p] = randomX;
        pallinoY[p] = randomY;
        pallinoW[p] = randomW;
        pallinoT[p] = randomT;
        pallinoTime[p] = random(0,10);
        pallinoDirection[p] = -1;
        pallinoSpeed[p] = random(5,10);
        pallinoColor[p] = 100;
    }
}

// RESETTA NEVE
void reset_neve() {
    for(int p=0;p<=45;p++) {
        //create random coords
        int randomX = random(0, width);
        int randomY = random(-30, -50);
        int randomW = random(8,14);
        int randomT = random(20,255);

        // store data
        neveX[p] = randomX;
        neveY[p] = randomY;
        neveW[p] = randomW;
        neveT[p] = randomT;
        neveTime[p] = random(0,10);
        neveDirection[p] = -1;
        neveSpeed[p] = random(5,10);
    }
}

// RESETTA QUADRI E RISPOSTE
void reset_paintings() {
    icons_painting[0][0] = 0;
    icons_painting[0][1] = 0;
    icons_painting[0][2] = 1;
    icons_painting[0][3] = 1;
    icons_painting[0][4] = 0;
    icons_painting[0][5] = 1;
}

```

```

icons_painting[0][6] = 1;
icons_painting[0][7] = 0;
icons_painting[0][8] = 1;

icons_painting[1][0] = 0;
icons_painting[1][1] = 0;
icons_painting[1][2] = 0;
icons_painting[1][3] = 0;
icons_painting[1][4] = 1;
icons_painting[1][5] = 1;
icons_painting[1][6] = 0;
icons_painting[1][7] = 1;
icons_painting[1][8] = 1;

icons_painting[2][0] = 0;
icons_painting[2][1] = 0;
icons_painting[2][2] = 0;
icons_painting[2][3] = 1;
icons_painting[2][4] = 0;
icons_painting[2][5] = 1;
icons_painting[2][6] = 1;
icons_painting[2][7] = 0;
icons_painting[2][8] = 1;

icons_painting[3][0] = 0;
icons_painting[3][1] = 0;
icons_painting[3][2] = 0;
icons_painting[3][3] = 1;
icons_painting[3][4] = 0;
icons_painting[3][5] = 1;
icons_painting[3][6] = 0;
icons_painting[3][7] = 1;
icons_painting[3][8] = 0;

for(i=0;i<=3;i++) {
    paintings_found[i] = 0;
    paintings_wrong[i] = 0;
    given_answers[i] = 0;
}
}

// VARIABLES //////////////////////////////////////
int countseconds = 0;
int i;
int j;
int k;
int p;
int totPunti = 0;           // score
int clues = 0;             // "there are XXX clues in this painting"
int distance = 0;          // cursor movement
int modeX = 35;            // starting point animation menu single/time/multi
int modeYquestion = 72;    // starting point animation text in final question
int modeXbarra = 0;        // starting point animation scrolling bar
int modeXicon = -150;      // starting point animation icons in the final question
int X_story = 240;         // starting point animation story_01
int X_story_2 = 240;       // starting point animation story_02
int Xbarchetta = -50;      // starting point animation barchetta (story_02)
int scrollingText = 0;     // how many time i can press UP/DOWN to scroll the final question
text
String letter = "";
String lastletter = "";
String nickname = "";
int immagineIndizio;
char lastkey;
Spook spook;              // class Spook

```

```

emptyImage emptyImage; // class Spook
String moodSpook;
boolean activeMenu = false;
boolean viewMenu = false;

// coriandoli
int[] pallinoX = new int[0];
int[] pallinoY = new int[0];
int[] pallinoW = new int[0];
int[] pallinoT = new int[0];
int[] pallinoTime = new int[0];
int[] pallinoDirection = new int[0];
int[] pallinoSpeed = new int[0];
int[] pallinoColor = new int[0];

// neve
int[] neveX = new int[0];
int[] neveY = new int[0];
int[] neveW = new int[0];
int[] neveT = new int[0];
int[] neveTime = new int[0];
int[] neveDirection = new int[0];
int[] neveSpeed = new int[0];
int[] neveColor = new int[0];

// MENU animation: starting positions
int POSITION_SINGLE = 35;
int POSITION_TIME = -207;
int POSITION_MULTY = -447;
int easeSpeed = 5;

// STORY animation: starting positions
int POSITION_VENICE = -40;
int POSITION_GHOST = -320;
int POSITION_MUSEUM = 0;
int POSITION_KHAN = -240;
int POSITION_PHONE = -490;

// FINAL QUESTION: starting position of the icons
int POSITION_SKULL = 100;
int POSITION_FOOD = 36;
int POSITION_SHIP = -25;
int POSITION_WEAPON = -89;
int POSITION_JEWEL = -155;
int POSITION_PLANT = -218;
int POSITION_FLAG = -282;
int POSITION_INSTRUMENT = -345;
int POSITION_ANIMAL = -413;

// FINAL QUESTION: starting position of the text
int POSITION_QUESTION_1 = 23;
int POSITION_QUESTION_2 = -30;
int POSITION_QUESTION_3 = -104;
int POSITION_QUESTION_4 = -136;
int POSITION_QUESTION_5 = -224;
int POSITION_QUESTION_6 = -275;

// STARTING millisec per le foto
int STARTING_QUESTION_1;
int STARTING_QUESTION_2;
int STARTING_QUESTION_3;
int STARTING_QUESTION_4;
int STARTING_QUESTION_5;

// GALLERY MUSEUMS animation: starting positions
int POSITION_ACCADEMIA = 58;
int POSITION_CORRER = -180;
int POSITION_CAPESARO = -420;

```

```
// GALLERY PICTURES animation: starting positions
int POSITION_SHIP_PIC = 58;
int POSITION_FLAG_PIC = -180;
int POSITION_PLANT_PIC = -420;
```

```
// IMAGES //////////////////////////////////////
```

```
PImage cornice_storta;
PImage exclamation_mark;
PImage final_mask;
PImage final_mask_testo;
PImage final_icone;
PImage final_selection;
PImage final_skull;
PImage final_food;
PImage final_jewel;
PImage final_plant;
PImage final_weapon;
PImage final_animal;
PImage final_ship;
PImage final_instrument;
PImage final_flag;
PImage final_screen;
PImage frecce_sx;
PImage frecce_dx;
PImage fumini;
PImage game_mode;
PImage griglia_icone;
PImage i_animal;
PImage i_animal_piccolo;
PImage i_flag;
PImage i_flag_piccolo;
PImage i_food;
PImage i_food_piccolo;
PImage i_instrument;
PImage i_instrument_piccolo;
PImage i_jewel;
PImage i_jewel_piccolo;
PImage i_plant;
PImage i_plant_piccolo;
PImage i_ship;
PImage i_ship_piccolo;
PImage i_skull;
PImage i_skull_piccolo;
PImage i_weapon;
PImage i_weapon_piccolo;
PImage icona_stanza;
PImage icona_quadro;
PImage icona_spook;
PImage imageBackground;
PImage logo;
PImage linea_arancione;
PImage museo;
PImage nuvola_piccola;
PImage nuvola_grande;
PImage pop_up_nome;
PImage quadro02;
PImage quadro03;
PImage quadro05;
PImage quadro06;
PImage quadro08;
PImage quadro14;
PImage quadro15;
PImage quadro17;
PImage quadro18;
PImage quadro23;
PImage quadro25;
PImage quadro26;
PImage quadro28;
PImage quadro33;
```

```

PImage quadro35;
PImage quadro37;
PImage room_complete;
PImage selezione_grande_1;
PImage selezione_grande_2;
PImage select_1_1;
PImage select_1_2;
PImage select_1_3;
PImage select_2_1;
PImage select_2_2;
PImage select_2_3;
PImage select_3_1;
PImage select_3_2;
PImage select_3_3;
PImage sfondo;
PImage spook_felice;
PImage spook_triste;
PImage spook_esulta;
PImage spook_paura;
PImage spook_saluta;
PImage spook_racconta;
PImage spook_giravolta;
PImage spook_occhei;
PImage spook_piange;
PImage story_01;
PImage story_02;
PImage story_03;

```

```

PImage gallery_musei;
PImage gallery_foto;
PImage icona_museo;

```

```

// FONTS //////////////////////////////////////

```

```

PFont din_black_12; // small text bold -> section titles & softkeys
menu
PFont din_black_12_brown; // -> painting titles (list)
PFont din_black_12_white; // -> selected painting titles
(list)
PFont din_light_12_light_brown; // small text light-> application name + score
PFont din_light_12; // -> section subtitles
PFont din_light_12_brown; // -> painting authors (list)
PFont din_light_12_white; // -> selected painting authors
(list)
PFont din_light_15_white; // big text light -> "application goes stand-by"
PFont din_light_15_brown; // -> normal text
PFont din_light_15_yellow; // -> pop-up text
PFont din_black_15; // big text bold -> painting authors
PFont din_black_15_brown; // big text bold -> "the end" in final questio

```

```

// COLORS //////////////////////////////////////

```

```

color brown;
color light_brown;
color orange;
color white;
color yellow;
color black;
color dark_yellow;

```

```

// CLUES & PAITINGS //////////////////////////////////////

```

```

//// each line is a painting
//// each column is a true or false clue
int icons_painting[][] = {

```

```

{
    0,0,1,1,0,1,1,0,1
},
{

```

```

    0,0,0,0,1,1,0,1,1          }
    {
    0,0,0,1,0,1,1,0,1          }
    {
    0,0,0,1,0,1,0,1,0          }
};

int paintings_found[] = {      // found paintings (1) cannot be selected
    0,0,0,0};

int paintings_wrong[] = {     // found paintings (1) cannot be selected
    0,0,0,0};

int correct_answers[] = {    // correct answers to final question
    0,7,2,6,3};

int given_answers[] = {      // given answers to final question
    0,0,0,0,0};              // 0 -> not answered yet ///// 1-> wrong answer //// 2
-> correct

String titolo_quadro[] = {
    "L'ARRIVO DEI PELLEGRINI", "SANTA VENERANDA IN TRONO", "APOTEOSI DEI DIECIMILA MARTIRI", "PALA
DI SAN GIOBBE"};
String autore_quadro[] = {
    "Carpaccio (1490)", "Bastiani (1470)", "Carpaccio (1515)", "Bellini (1487)"};

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//  MODES  //////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

/// NAMING GAME-MODES
int MODE_SINGLE_PLAYER = 1;
int MODE_TIME_CHALLENGE = 2;
int MODE_MULTIPLAYER = 3;
int MODE_GALLERY = 4;

// NAMING SECTIONS
int START_SCREEN = 0;
int REGISTRATION = 1;
int STORY_01 = 2;
int PLAY_OR_GALLERY = 15;
int MUSEUM_LOCALIZATION = 3;
int MENU_GAME_MODE = 4;
int NARRATIVE = 5;
int MUSEUM_ROOM = 6;
int ICONS = 7;
int TAKE_PHOTO = 8;
int GOT_CLUES = 9;
int ROOM_COMPLETE = 10;
int STAND_BY_TO_EXIT = 11;
int EXIT_MUSEUM = 12;
int FINAL_QUESTION = 13;
int LAST_SCREEN = 14;
int TIME_IS_UP = 20;          // Time mode
int MUSEUM_SELECTION = 30;
int PICS_BROWSING = 31;

//during game focus (used for menu animation)
int gameModeFocus = MODE_SINGLE_PLAYER;

```



```
// NAMING MODES
/// start screens
int LOGO_SCREEN = 0;
int LOGIN = 1;
int REGISTRATION_POPUP = 2;

// story01
int HI_IM_SPOOK = 0;
int IM_DETECTIVE = 1;
int HELP_GHOST = 2;
int REST_IN_PEACE = 3;
int WOULD_YOU_HELP = 4;

/// narrative
int IN_THIS_MUSEUM = 0;
int MARCO_POLO_LEFT = 1;
int RECOGNIZE_CLUES = 2;
int AT_THE_END = 3;
int DONT_GET_WRONG = 4;
int READY = 5;
int ENTER_A_ROOM = 6;
int STAND_BY = 7;

/// museum_room
int PAINT_1 = 0;
int PAINT_2 = 1;
int PAINT_3 = 2;
int PAINT_4 = 3;

/// icons
int SKULL = 0;
int FOOD = 1;
int SHIP = 2;
int WEAPON = 3;
int JEWEL = 4;
int PLANT = 5;
int FLAG = 6;
int INSTRUMENT = 7;
int ANIMAL = 8;

/// final question
int QUESTION_1 = 0;
int QUESTION_2 = 1;
int QUESTION_3 = 2;
int QUESTION_4 = 3;
int QUESTION_5 = 4;
int QUESTION_6 = 5;

/// last screen
int CASE_SOLVED = 0;
int GAME_COMPLETE = 1;

/// menu
int BACK_TO_GAME = 0;
int GALLERY = 1;
int RESTART_THE_GAME = 2;
int EXIT = 4;

/// game or play
int PLAY_THE_GAME = 0;
int VIEW_GALLERY = 1;

/// time
int TIME = 8;
int TIME_READY = 9;
int TIME_ENTER_A_ROOM = 10;
```

```

/// museum selection
int ACCADEMIA = 0;
int CORRER = 1;
int CAPESARO = 2;

/// picture browsing
int SHIP_PIC = 0;
int FLAG_PIC = 1;
int PLANT_PIC = 2;

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
// State, Information Collected From Use (Variables)
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

int sezione = START_SCREEN;           // starting SECTION
int mode = LOGO_SCREEN;                 // starting FOCUS
int modeMenu = BACK_TO_GAME;           // starting MENU
int modeIcon = 0;                       // starting ICON
int gameMode = 0;                       // starting GAME MODE

int museumSelection = 0;                // starting MUSEUM SELECTION
int picsBrowse = 0;                     // starting PICTURE BROWSING

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
// FUNCTIONS
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

// LOAD IMAGES
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

void loadImages() {
    cornice_storta = loadImage("cornice_storta.png");
    exclamation_mark = loadImage("exclamation_mark.png");
    final_mask = loadImage("final_mask.png");
    final_mask_testo = loadImage("final_mask_testo.png");
    final_icone = loadImage("final_icone.png");
    final_selection = loadImage("final_selection.png");
    final_skull = loadImage("final_skull.gif");
    final_food = loadImage("final_food.gif");
    final_ship = loadImage("final_ship.gif");
    final_plant = loadImage("final_plant.gif");
    final_jewel = loadImage("final_jewel.gif");
    final_weapon = loadImage("final_weapon.gif");
    final_flag = loadImage("final_flag.gif");
    final_instrument = loadImage("final_instrument.gif");
    final_animal = loadImage("final_animal.gif");
    final_screen = loadImage("final_screen.png");
    frecce_sx = loadImage("frecce_sx.png");
    frecce_dx = loadImage("frecce_dx.png");
    fumini = loadImage("fumini.png");
    griglia_icone = loadImage("griglia_icone.png");
    i_animal = loadImage("i-animal.png");
    i_animal_piccolo = loadImage("i-animal-piccolo.png");
    i_flag = loadImage("i-flag.png");
    i_flag_piccolo = loadImage("i-flag-piccolo.png");
    i_food = loadImage("i-food.png");
    i_food_piccolo = loadImage("i-food-piccolo.png");
    i_instrument = loadImage("i-instrument.png");
    i_instrument_piccolo = loadImage("i-instrument-piccolo.png");
    i_jewel = loadImage("i-jewel.png");
    i_jewel_piccolo = loadImage("i-jewel-piccolo.png");
    i_plant = loadImage("i-plant.png");
    i_plant_piccolo = loadImage("i-plant-piccolo.png");
    i_ship = loadImage("i-ship.png");
    i_ship_piccolo = loadImage("i-ship-piccolo.png");
}

```

```

i_weapon = loadImage("i-weapon.png");
i_weapon_piccolo = loadImage("i-weapon-piccolo.png");
i_skull = loadImage("i-skull.png");
i_skull_piccolo = loadImage("i-skull-piccolo.png");
icona_stanza = loadImage("icona_stanza.png");
icona_quadro = loadImage("icona_quadro.png");
icona_spook = loadImage("icona_spook.png");
imageBackground = loadImage("imageBackground.png");
logo = loadImage("logo.png");
linea_arancione = loadImage("linea_arancione.png");
game_mode = loadImage("game-mode.png");
museo = loadImage("museo.png");
nuvola_piccola = loadImage("nuvola-piccola.png");
nuvola_grande = loadImage("nuvola-grande.png");
pop_up_nome = loadImage("pop_up_nome.png");
quadro02 = loadImage("quadro02.png");
quadro03 = loadImage("quadro03.png");
quadro05 = loadImage("quadro05.png");
quadro06 = loadImage("quadro06.png");
quadro08 = loadImage("quadro08.png");
quadro14 = loadImage("quadro14.png");
quadro15 = loadImage("quadro15.png");
quadro17 = loadImage("quadro17.png");
quadro18 = loadImage("quadro18.png");
quadro23 = loadImage("quadro23.png");
quadro25 = loadImage("quadro25.png");
quadro26 = loadImage("quadro26.png");
quadro28 = loadImage("quadro28.png");
quadro33 = loadImage("quadro33.png");
quadro35 = loadImage("quadro35.png");
quadro37 = loadImage("quadro37.png");
room_complete = loadImage("room_complete.png");
select_1_1 = loadImage("select_1.1.png");
select_1_2 = loadImage("select_1.2.png");
select_1_3 = loadImage("select_1.3.png");
select_2_1 = loadImage("select_2.1.png");
select_2_2 = loadImage("select_2.2.png");
select_2_3 = loadImage("select_2.3.png");
select_3_1 = loadImage("select_3.1.png");
select_3_2 = loadImage("select_3.2.png");
select_3_3 = loadImage("select_3.3.png");
selezione_grande_1 = loadImage("selezione_grande_1.png");
selezione_grande_2 = loadImage("selezione_grande_2.png");
spook_felice = loadImage("spook-felice.png");
spook_triste = loadImage("spook-triste.png");
spook_esulta = loadImage("spook-esulta.png");
spook_saluta = loadImage("spook-saluta.png");
spook_racconta = loadImage("spook-racconta.png");
spook_paura = loadImage("spook-paura.png");
spook_giravolta = loadImage("spook-giravolta.png");
spook_occhei = loadImage("spook-occhei.png");
spook_piange = loadImage("spook-piange.png");
sfondo = loadImage("sfondo.png");
story_01 = loadImage("story_01.png");
story_02 = loadImage("story_02.png");
story_03 = loadImage("story_03.png");

gallery_musei = loadImage("gallery_musei.png");
gallery_foto = loadImage("gallery_foto.png");
icona_museo = loadImage("icona_museo.png");
}

// LOAD FONTS //////////////////////////////////////

void loadFonts()
{
    // menu fonts
    din_black_12 = loadFont("din_black_12.mvlw", orange);
    din_black_12_brown = loadFont("din_black_12.mvlw", brown);
}

```

```

din_black_12_white = loadFont("din_black_12.mvlw", white);
din_light_12 = loadFont("din_light_12.mvlw", orange);
din_light_12_light_brown = loadFont("din_light_12.mvlw", light_brown);
din_light_12_brown = loadFont("din_light_12.mvlw", brown);
din_light_12_white = loadFont("din_light_12.mvlw", white);
din_light_15_brown = loadFont("din_light_15.mvlw", brown);
din_light_15_yellow = loadFont("din_light_15.mvlw", yellow);
din_light_15_white = loadFont("din_light_15.mvlw", white);
din_black_15 = loadFont("din_black_15.mvlw", orange);
din_black_15_brown = loadFont("din_black_15.mvlw", brown);
}

// MOVEMENT FUNCION  //////////////////////////////////////

int easeTo(int currentOriginal, int targetOriginal, int speed)
{
    // make values large for math
    int current = currentOriginal * 1000; // make current value large for math (e.g. 2 becomes 2000)
    int target = targetOriginal * 1000; // make target value large for math (e.g. 33 becomes 33000)
    // do math to calculate our next value
    int change = target - current; // find out how much change there is (e.g. 33000 - 2000 = 31000)
    int changeLittle = change / speed; // make the change a little change (e.g. 31000 / 4 = 7750)
    int next = current + changeLittle; // change the current value a little (e.g. 2000 + 7750 =
9750)
    // make next value small for screen
    next = next / 1000;
    // if our little change was so little that we didn't move...
    if(next == currentOriginal){
        next = targetOriginal; // our next step is our target
    }
    return next; // return our next value (e.g. 9750 / 1000 = 9, remember that we started with 2)
}

// SELECTING PHOTOS (FINAL QUESTION)  //////////////////////////////////////
void selectPicture(int imgX, int imgY, int iconSelected, int j) {
    if(j==10) j=iconSelected;
    if(j==0) image(final_skull, imgX, imgY);
    if(j==1) image(final_food, imgX, imgY);
    if(j==2) image(final_ship, imgX, imgY);
    if(j==3) image(final_weapon, imgX, imgY);
    if(j==4) image(final_jewel, imgX, imgY);
    if(j==5) image(final_plant, imgX, imgY);
    if(j==6) image(final_flag, imgX, imgY);
    if(j==7) image(final_instrument, imgX, imgY);
    if(j==8) image(final_animal, imgX, imgY);
}

// CAMERA FUNCTION  //////////////////////////////////////

void showCamera(int x, int y, int w, int h, PImage imageBackground) {
    noLoop();
    myCapture.show(x,y,w,h);
    image(imageBackground, 0, 0);
}

void hideCamera() {
    myCapture.hide();
    loop();
}

void readCamera() {
    myCapture.read();
}

void destroy() {
    myCapture.close();
}

```

```

// TIME MODE VAR //////////////////////////////////////
int starTime = 0;           // start-playing-game time
int endTime = 0;           // start-playing-game time
int sizeX = 5;             // size rect
int sizeY = 15;           // size rect
int drawX = 0;             // x position
int drawY = 0;             // y position
int traslaY = 53;          // start y position
int sbaglio = 0;
int timeMax = 360000;      // room time

PImage cursore;

// TIME MODE FUNCTIONS //////////////////////////////////////
// TIME CHECK //////////////////////////////////////

void timeCheck(){
  if((millis() - starTime) > timeMax){
    sezione = TIME_IS_UP;
  }
}

void drawCursor(){
  int tempo = millis()-starTime;

  //image(cursore,236,0,sizeX,sizeY,0,42);

  drawX = (((tempo/1000)*2)/3);
  drawY = (traslaY - (((tempo/1000)*2)/45));
  println(drawX + " - " + drawY);
  fill(brown);
  rect(drawX, drawY, sizeX, sizeY);
}

// *****
// ** GALLERY *****
// *****

void drawGallery (int sezione, int mode) // sezione, sottosezione
{

  // SELECT MUSEUM
  *****
  if(sezione == MUSEUM_SELECTION) {

    drawSoftKeys(true,true,false);

    // titolo
    image(linea_arancione, 0, 46);
    textFont(din_black_12);
    textAlign(LEFT);
    text("SCEGLI UN MUSEO", 32, 34);

    // ACCADEMIA //////////////////////////////////////
    if(mode == ACCADEMIA) {
      // frecce
      image(frecce_dx, 220, 160);

      // image
      modeX = easeTo(modeX, POSITION_ACCADEMIA, easeSpeed);
      image(gallery_musei, modeX, 100);
    }
  }
}

```

```

    textFont(din_light_15_brown);
    text("Museo Ca Pesaro", 32, 87);
}

// CORRER //////////////////////////////////////
if(mode == CORRER) {
    // frecce
    image(frecce_sx, 5, 160);
    image(frecce_dx, 220, 160);

    //iamege
    modeX = easeTo(modeX, POSITION_CORRER, easeSpeed);
    image(gallery_musei, modeX, 100);

    textFont(din_light_15_brown);
    text("Museo Correr", 32, 87);
}

// CA PESARO //////////////////////////////////////
if(mode == CAPESARO) {
    // frecce
    image(frecce_sx, 5, 160);

    // image
    modeX = easeTo(modeX, POSITION_CAPESARO, easeSpeed);
    image(gallery_musei, modeX, 100);

    textFont(din_light_15_brown);
    text("Museo dell'Accademia", 32, 87);
}
}

// ICONS SCREEN
*****
if(sezione == ICONS) {

    // titolo
    image(linea_arancione, 0, 46);
    image(icona_museo, 6, 24);

    textAlign(LEFT);
    textFont(din_black_12);
    text("FOTOGALLERY", 32, 34);
    textFont(din_light_12);
    text("Museo Correr", 32, 46);

    //testo
    textFont(din_light_15_brown);
    text("Scegli un indizio per guardare le immagini collezionate:", 32, 72, 180, 150);

    // grafica icone
    image(griglia_icone, 0, 0);

    // SKULL
    if(modeIcon == SKULL) {
        // image
        image(select_1_1, 32, 114);
    }

    // FOOD
    if(modeIcon == FOOD) {
        // image
        image(select_1_2, 92, 122);
    }

    // SHIP
    if(modeIcon == SHIP) {
        // image
        image(select_1_3, 150, 122);
    }
}

```

```

}

// WEAPON
if(modeIcon == WEAPON) {
    // image
    image(select_2_1, 32, 182);
}

// JEWEL
if(modeIcon == JEWEL) {
    // image
    image(select_2_2, 92, 182);
}

// PLANT
if(modeIcon == PLANT) {
    // image
    image(select_2_3, 149, 182);
}

// FLAG
if(modeIcon == FLAG) {
    // image
    image(select_3_1, 32, 239);
}

// INSTRUMENT
if(modeIcon == INSTRUMENT) {
    // image
    image(select_3_2, 92, 239);
}

// ANIMAL
if(modeIcon == ANIMAL) {
    // image
    image(select_3_3, 149, 239);
}

// icone
image(i_skull, 46, 128);
image(i_food, 100, 129);
image(i_ship, 152, 127);
image(i_weapon, 37, 185);
image(i_jewel, 100, 185);
image(i_plant, 155, 186);
image(i_flag, 46, 242);
image(i_instrument, 100, 242);
image(i_animal, 153, 242);

drawSoftKeys(true,true,true);
}

// SELECT PICTURE
*****
if(sezione == PICS_BROWSING) {

    drawSoftKeys(true,false,true);

    // titolo
    image(linea_arancione, 0, 46);
    textFont(din_black_12);
    textAlign(LEFT);
    text("INDIZI COLLEZIONATI", 32, 34);

    // SHIP //////////////////////////////////////
    if(mode == SHIP_PIC) {
        // freccine

```

```

image(frecce_dx, 220, 160);

// image
modeX = easeTo(modeX, POSITION_SHIP_PIC, easeSpeed);
image(gallery_foto, modeX, 100);

textFont(din_black_12_brown);
text("PALA DI SAN GIOBBE", 32, 78);
textFont(din_light_12_brown);
text("Bellini (1487)", 32, 90);
textFont(din_light_12_brown);
text("Questi cespugli venivano usati dai soldati per mimetizzarsi durante l'assedio alla
città", 32, 250, 180, 60);
}

// FLAG ////////////////////////////////////////
if(mode == FLAG_PIC) {
// frecce
image(frecce_sx, 5, 160);
image(frecce_dx, 220, 160);

//iamge
modeX = easeTo(modeX, POSITION_FLAG_PIC, easeSpeed);
image(gallery_foto, modeX, 100);

textFont(din_black_12_brown);
text("SANTA VENERANDA IN TRONO", 32, 78);
textFont(din_light_12_brown);
text("Bastiani (1470)", 32, 90);
textFont(din_light_12_brown);
text("Il pino d'Aleppo è una pianta sempreverde che Marco Polo portò a Venezia dal Medio
Oriente", 32, 250, 180, 60);
}

// PLANT ////////////////////////////////////////
if(mode == PLANT_PIC) {
// frecce
image(frecce_sx, 5, 160);

// image
modeX = easeTo(modeX, POSITION_PLANT_PIC, easeSpeed);
image(gallery_foto, modeX, 100);

textFont(din_black_12_brown);
text("L'ARRIVO DEI PELLEGRINI", 32, 78);
textFont(din_light_12_brown);
text("Carpaccio (1490)", 32, 90);
textFont(din_light_12_brown);
text("Il nespolo è una pianta cinese, la preferita di Qubilay Khan", 32, 250, 180, 60);
}
}

// DRAW SOFTKEYS ////////////////////////////////////////
//// draws softkeys instruction (menu / OK / back) if true

void drawSoftKeys(boolean left, boolean center, boolean right) {
textFont(din_black_12);
if(left) {
activeMenu = true;
textAlign(LEFT);
text("ALTRO?", 10, 311);
}
}

```



```

else activeMenu = false;
if(center) {
    textAlign(CENTER);
    text("VAI", width/2, 311);
}
if(right) {
    textAlign(RIGHT);
    text("OPS!", 230, 311);
}
}

// *****
// ** START SCREENS *****
// *****

void drawScreens(int sezione, int mode) // sezione, sottosezione
{
    textAlign(LEFT);

    // START SCREEN
    *****
    if(sezione == LOGO_SCREEN) {

        // LOGO_SCREEN //////////////////////////////////////
        background(0);
        image(logo, 0, 0);
    }

    // REGISTRATION
    *****
    if(sezione == REGISTRATION) {

        // screen
        // spook
        spook = new Spook("saluta", frameCount, true);

        drawSoftKeys(false,true,false);

        // text
        image(nuvola_piccola, 12, 75);
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("Ciao!", 32, 104, 150, 40);
        text("Come ti chiami?", 32, 120, 150, 40);

        // nickname
        textFont(din_black_15);
        textAlign(LEFT);
        text(nickname, 32, 150);

        // cursor distance //////////////////////////////////////
        distance = 0;
        for(i=0; i<=multitapBufferLength; i++) {
            if(multitapBuffer[i] == 'i' || multitapBuffer[i] == 'l') distance = distance + 4;
            else if(multitapBuffer[i] == 'j') distance = distance + 5;
            else if(multitapBuffer[i] == 'f' || multitapBuffer[i] == 't') distance = distance + 6;
            else if(multitapBuffer[i] == 'r' || multitapBuffer[i] == 's' || multitapBuffer[i] == 'I')
distance = distance + 7;
            else if(multitapBuffer[i] == 'c' || multitapBuffer[i] == 'z') distance = distance + 8;
            else if(multitapBuffer[i] == 'a' || multitapBuffer[i] == 'e' || multitapBuffer[i] == 'k' ||
multitapBuffer[i] == 'v' || multitapBuffer[i] == 'x' || multitapBuffer[i] == 'y' ||
multitapBuffer[i] == 'F' || multitapBuffer[i] == 'J' || multitapBuffer[i] == 'L') distance =
distance + 9;
            else if(multitapBuffer[i] == 'A' || multitapBuffer[i] == 'B' || multitapBuffer[i] == 'C' ||
multitapBuffer[i] == 'D' || multitapBuffer[i] == 'G' || multitapBuffer[i] == 'H' ||
multitapBuffer[i] == 'K' || multitapBuffer[i] == 'N' || multitapBuffer[i] == 'R' ||

```

```

multitapBuffer[i] == 'U' || multitapBuffer[i] == 'V' || multitapBuffer[i] == 'Y') distance =
distance + 11;
    else if(multitapBuffer[i] == 'w' || multitapBuffer[i] == 'N' || multitapBuffer[i] == 'Q')
distance = distance + 12;
    else if(multitapBuffer[i] == 'M') distance = distance + 13;
    else if(multitapBuffer[i] == 'W') distance = distance + 15;
    else if(multitapBuffer[i] == 'w') distance = distance + 16;
    else distance = distance + 10;
}

if(mode == REGISTRATION) {
    // cursor: appears only every 1/2 sec
    strokeWeight(2);
    stroke(brown);
    //if(frameCount%20<=10) line(32+(10*(nickname.length())), 152, 42+(10*(nickname.length())),
152);
    if(frameCount%20<=10) line(22+distance, 152, 32+distance, 152);
}

// POP-UP ////////////////////////////////////////
else if(mode == REGISTRATION_POPUP) {

    // ALERT nickname non inserito
    if(nickname.length() == 0) {
        drawSoftKeys(false, false, true);
        // popup
        image(pop_up_nome, 3, 100);
        textFont(din_light_15_yellow);
        textAlign(LEFT);
        text("Cosa?? Non hai un nome? Com'è possibile?!?", 15, 127, 200, 70);
    }
    else {

        drawSoftKeys(false, true, true);

        // popup
        image(pop_up_nome, 3, 100);
        textFont(din_light_15_yellow);
        textAlign(LEFT);
        text("Allora ti chiami "+nickname+"?", 15, 137, 190, 40);
    }
}

// titolo
image(linea_arancione, 0, 46);
textFont(din_black_12);
textAlign(LEFT);
text("PRESENTAZIONI", 32, 34);
image(icona_spook, 6, 24);
}

// STORY_01 *****
if(sezione == STORY_01) {
    moodSpook = "saluta";
    image(fumini,0,0);

    // HI IM SPOOK ////////////////////////////////////////
    if(mode == HI_IM_SPOOK) {
        // text
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("Ciao "+nickname+", io sono Spook.", 32, 72, 190, 70);
    }
}

```

```

// IM A DETECTIVE //////////////////////////////////////
if(mode == IM_DETECTIVE) {
    moodSpook = "saluta";
    // text
    textFont(din_light_15_brown);
    textAlign(LEFT);
    text("Ciao "+nickname+", io sono Spook.", 32, 72, 190, 70);
    text("Facevo il detective nell'antica Repubblica di Venezia.", 32, 102, 175, 70);

    // image slide: venice skyline
    X_story = easeTo(X_story, POSITION_VENICE, easeSpeed);
    image(story_01, X_story, 0);
}

// HELP GHOSTS //////////////////////////////////////
if(mode == HELP_GHOST) {
    moodSpook = "saluta";
    // text
    textFont(din_light_15_brown);
    textAlign(LEFT);
    text("Ora aiuto gli altri fantasmi a scoprire cosa gli accadde in vita.", 32, 72, 180, 150);

    // image slide
    X_story = easeTo(X_story, POSITION_GHOST, easeSpeed);
    image(story_01, X_story, 0);
}

// REST_IN_PEACE //////////////////////////////////////
if(mode == REST_IN_PEACE) {
    moodSpook = "saluta";
    // text
    textFont(din_light_15_brown);
    textAlign(LEFT);
    text("Ora aiuto gli altri fantasmi a scoprire cosa gli accadde in vita.", 32, 72, 180, 150);
    text("Non riposeranno in pace finchè non ricorderanno il loro passato.", 32, 127, 175, 150);

    // image slide
    X_story = easeTo(X_story, POSITION_GHOST, easeSpeed);
    image(story_01, X_story, 0);
}

// WOULD YOU HELP //////////////////////////////////////
if(mode == WOULD_YOU_HELP) {
    moodSpook = "racconta";
    // text
    textFont(din_light_15_brown);
    textAlign(LEFT);
    text("Ora aiuto gli altri fantasmi a scoprire cosa gli accadde in vita.", 32, 72, 180, 150);
    text("Non riposeranno in pace finchè non ricorderanno il loro passato.", 32, 127, 175, 150);
    text("Mi daresti una mano?", 32, 179, 130, 150);

    // image slide
    X_story = easeTo(X_story, -500, easeSpeed);
    image(story_01, X_story, 0);
}

// spook
spook = new Spook(moodSpook, frameCount, false);

// titolo
image(linea_arancione, 0, 46);
textFont(din_black_12);
textAlign(LEFT);
text("INTRODUZIONE", 32, 34);
image(icona_spook, 6, 24);

drawSoftKeys(false,true,false);
}

```

```

// PLAY_OR_GALLERY //////////////////////////////////////
if(sezione == PLAY_OR_GALLERY) {

    // game options
    // PLAY //////////////////////////////////////
    if(mode==PLAY_THE_GAME) {
        image(selezione_grande_1, 3, 190);
        textFont(din_black_12_white);
    }
    else {
        textFont(din_black_12_brown);
    }
    text("GIOCHIAMO", 32, 217);

    // EXIT //////////////////////////////////////
    if(mode==VIEW_GALLERY) {
        image(selezione_grande_2, 3, 236);
        textFont(din_black_12_white);
    }
    else {
        textFont(din_black_12_brown);
    }
    text("FOTOGALLERY", 32, 262);

    // titolo
    image(linea_arancione, 0, 46);
    textFont(din_black_12);
    textAlign(LEFT);
    text("MENU DI GIOCO", 32, 34);
    image(icona_spook, 6, 24);

    // spook
    spook = new Spook("racconta", frameCount, false);

    // text
    image(nuvola_piccola, 12, 75);
    textFont(din_light_15_brown);
    textAlign(LEFT);
    text("Bene, "+nickname+"... Cosa vuoi fare ora?", 32, 114, 160, 40);

    drawSoftKeys(false,true,false);
}

// MUSEUM LOCALIZATION
*****
if(sezione == MUSEUM_LOCALIZATION) {
    // titolo
    image(linea_arancione, 0, 46);
    textFont(din_black_12);
    textAlign(LEFT);
    text("GALLERIA DELL'ACCADEMIA", 32, 34);

    // screen
    textFont(din_light_15_brown);
    textAlign(LEFT);
    text(nickname+"", sei nella Galleria dell'Accademia", 32, 240, 180, 60);
    image(museo, width/2-63, 80);

    drawSoftKeys(false,true,false);
}

// SELECT GAME MODE
*****
if(sezione == MENU_GAME_MODE) {

    drawSoftKeys(true,true,false);
}

```

```

// titolo
image(linea_arancione, 0, 46);
textFont(din_black_12);
textAlign(LEFT);
text("SCEGLI IL LIVELLO", 32, 34);

// SINGLE PLAYER //////////////////////////////////////
if(mode == MODE_SINGLE_PLAYER) {
    // freccine
    image(frecce_dx, 220, 160);

    // image
    modeX = easeTo(modeX, POSITION_SINGLE, easeSpeed);
    image(game_mode, modeX, 100);

    textFont(din_light_15_brown);
    text("Un giocatore", 32, 87);
}

// TIME CHALLENGE //////////////////////////////////////
if(mode == MODE_TIME_CHALLENGE) {
    // freccine
    image(frecce_sx, 5, 160);
    image(frecce_dx, 220, 160);

    //image
    modeX = easeTo(modeX, POSITION_TIME, easeSpeed);
    image(game_mode, modeX, 100);

    textFont(din_light_15_brown);
    text("Sfida a tempo", 32, 87);
}

// MULTIPLAYER //////////////////////////////////////
if(mode == MODE_MULTIPLAYER) {
    // freccine
    image(frecce_sx, 5, 160);

    // image
    modeX = easeTo(modeX, POSITION_MULTY, easeSpeed);
    image(game_mode, modeX, 100);

    textFont(din_light_15_brown);
    text("Più giocatori", 32, 87);
}
}

// *****
// ** MUTLIPLAYER *****
// *****

void drawMultiplayer (int sezione, int mode) // sezione, sottosezione
{
}

// *****
// ** MENU *****
// *****

void drawMenu (int mode) {

```

```

drawSoftKeys(false,true,false);
image(fumini,0,0);

// titolo
/*image(linea_arancione, 0, 46);
textFont(din_black_12);
textAlign(LEFT);
text("MENU", 32, 34);
*/
textAlign(CENTER);

pushMatrix();
translate(0,-30);

// BACK TO GAME //////////////////////////////////////
if(modeMenu==BACK_TO_GAME) {
  // selezione
  image(selezione_grande_1, 6, 106);
  textFont(din_black_12_white);
}
else {
  textFont(din_black_12_brown);
}
text("TORNA AL GIOCO", 120, 133);

// GALLERY //////////////////////////////////////
if(modeMenu==GALLERY) {
  image(selezione_grande_2, 6, 153);
  textFont(din_black_12_white);
}
else {
  textFont(din_black_12_brown);
}
text("FOTOGALLERY", 120, 178);

// RESTART GAME //////////////////////////////////////
if(modeMenu==RESTART_THE_GAME) {
  image(selezione_grande_1, 6, 196);
  textFont(din_black_12_white);
}
else {
  textFont(din_black_12_brown);
}
text("RICOMINCIA IL GIOCO", 120, 223);

// EXIT //////////////////////////////////////
if(modeMenu==EXIT) {
  image(selezione_grande_2, 6, 242);
  textFont(din_black_12_white);
}
else {
  textFont(din_black_12_brown);
}
text("ESCI", 120, 268);
popMatrix();
}

// *****
// ** SINGLE PLAYER *****
// *****

void drawSinglePlayer (int sezione, int mode) // sezione, sottosezione
{

```

```

// NARRATIVE *****
if(sezione == NARRATIVE) {

    image(fumini,0,0);

    // IN_THIS_MUSEUM //////////////////////////////////////
    if(mode == IN_THIS_MUSEUM) {
        moodSpook = "saluta";           // changes spook's mood in the spook animation

        // text
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("In questo museo ci sono degli indizi sullo spirito di Hao Dong, figlia di Qubilay Khan
e moglie di Marco Polo.", 32, 72, 180, 200);

        // image slide: museum
        X_story_2 = easeTo(X_story_2, POSITION_MUSEUM, easeSpeed);
        image(story_02, X_story_2, 0);
    }

    // SIGNORA MARCO POLO //////////////////////////////////////
    if(mode == MARCO_POLO_LEFT) {
        moodSpook = "triste";

        // text
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("Le hanno fatto credere che Marco Polo l'abbia abbandonata per sempre. Lei non riesce a
farsene una ragione... Ma sono sicuro che c'è qualcosa sotto!", 32, 72, 180, 200);

        // barchetta che passa
        if(Xbarchetta <= 300) Xbarchetta = Xbarchetta + 1;
        image(story_03, Xbarchetta, 208);

        // image slide: signora marco polo
        X_story_2 = easeTo(X_story_2, POSITION_KHAN, easeSpeed);
        image(story_02, X_story_2, 0);
    }

    // RECOGNIZE_CLUES //////////////////////////////////////
    if(mode == RECOGNIZE_CLUES) {
        moodSpook = "felice";

        // image slide: mobile phone
        //X_story_2 = easeTo(X_story_2, POSITION_PHONE, easeSpeed);
        //image(story_02, X_story_2, 0);

        // text
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("Trova gli indizi nascosti nei quadri del museo.", 32, 72, 180, 150);
    }

    // AT_THE_END //////////////////////////////////////
    if(mode == AT_THE_END) {
        moodSpook = "felice";

        // image slide
        //X_story_2 = easeTo(X_story_2, -700, easeSpeed);
        //image(story_02, X_story_2, 0);

        // text
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("Trova gli indizi nascosti nei quadri del museo.", 32, 72, 180, 150);
        text("Alla fine cercheremo di ricostruire la storia della povera Hao Dong.", 32, 127, 175,
150);

```

```

}

// DONT_GET_WRONG //////////////////////////////////////
if(mode == DONT_GET_WRONG) {
    moodSpook = "paura";

    // text
    textFont(din_light_15_brown);
    textAlign(LEFT);
    text("Attenzione a non sbagliare, però, altrimenti non potremo più usare il quadro come
prova!", 32, 72, 180, 150);

    // image exclamation mark
    image(exclamation_mark, 86, 160);
}

// READY //////////////////////////////////////
if(mode == READY) {
    moodSpook = "racconta";

    // text
    textFont(din_light_15_brown);
    textAlign(LEFT);
    text("Attenzione a non sbagliare, però, altrimenti non potremo più usare il quadro come
prova!", 32, 72, 180, 150);
    text("Siamo pronti?", 32, 155, 175, 150);
}

// ENTER A ROOM //////////////////////////////////////
if(mode == ENTER_A_ROOM) {
    moodSpook = "felice";

    // text
    textFont(din_light_15_brown);
    textAlign(LEFT);
    text("Attenzione a non sbagliare, però, altrimenti non potremo più usare il quadro come
prova!", 32, 72, 180, 150);
    text("Siamo pronti?", 32, 155, 175, 150);
    text("Entra nella prima stanza per iniziare il gioco.", 32, 177, 110, 150);
}

// drawing spook
spook = new Spook(moodSpook, frameCount, false);

// titolo
image(linea_arancione, 0, 46);
textFont(din_black_12);
textAlign(LEFT);
text("INTRODUZIONE", 32, 34);
image(icona_spook, 6, 24);

drawSoftKeys(false,true,false);

// STAND_BY //////////////////////////////////////
if(mode == STAND_BY) {
    background(0); // black background
    // text
    textFont(din_light_15_white);
    textAlign(LEFT);
    text("L'applicazione va in stand-by finchè il bambino non entra in una stanza.", 32, 147,
175, 150);
}
}

```



```

// MUSEUM_ROOM
*****
if(sezione == MUSEUM_ROOM) {

    drawSoftKeys(true,true,false);

    // titolo
    image(linea_arancione, 0, 46);
    textFont(din_black_12);
    textAlign(LEFT);
    text("STANZA 14", 32, 34);
    image(icona_stanza, 6, 24);

    // testo
    textFont(din_light_15_brown);
    text("Guardati in giro e trova uno di questi quadri...", 32, 72, 190, 70);

    // paintings list
    // 1 //////////////////////////////////////
    if(mode==PAINT_1) {
        // selezione
        image(selezione_grande_1, 3, 106);
        textFont(din_black_12_white);
    }
    else {
        textFont(din_black_12_brown);
    }
    if(paintings_found[PAINT_1] == 0) {
        image(icona_quadro, 6, 117);
    }
    text("L'ARRIVO DEI PELLEGRINI", 32, 128);

    if(mode==PAINT_1) {
        textFont(din_light_12_white);
    }
    else {
        textFont(din_light_12_brown);
    }
    text("Carpaccio (1490)", 32, 140);

    // 2 //////////////////////////////////////
    if(mode==PAINT_2) {
        image(selezione_grande_2, 3, 153);
        textFont(din_black_12_white);
    }
    else {
        textFont(din_black_12_brown);
    }
    if(paintings_found[PAINT_2] == 0) {
        image(icona_quadro, 6, 162);
    }
    text("SANTA VENERANDA IN TRONO", 32, 173);
    if(mode==PAINT_2) {
        textFont(din_light_12_white);
    }
    else {
        textFont(din_light_12_brown);
    }
    text("Bastiani (1470)", 32, 185);

    // 3 //////////////////////////////////////
    if(mode==PAINT_3) {
        image(selezione_grande_1, 3, 196);
        textFont(din_black_12_white);
    }
    else {
        textFont(din_black_12_brown);
    }
    if(paintings_found[PAINT_3] == 0) {

```

```

    image(icona_quadro, 6, 207);
}
text("APOTEOSI DEI DIECIMILA MARTIRI", 32, 218);
if(mode==PAINT_3) {
    textFont(din_light_12_white);
}
else {
    textFont(din_light_12_brown);
}
text("Carpaccio (1515)", 32, 230);

// 4 //////////////////////////////////////
if(mode==PAINT_4) {
    image(selezione_grande_2, 3, 242);
    textFont(din_black_12_white);
}
else {
    textFont(din_black_12_brown);
}
if(paintings_found[PAINT_4] == 0) {
    image(icona_quadro, 6, 252);
}
text("PALA DI SAN GIOBBE", 32, 263);
if(mode==PAINT_4) {
    textFont(din_light_12_white);
}
else {
    textFont(din_light_12_brown);
}
text("Bellini (1487)", 32, 275);
}

```

```
// ICONS SCREEN
```

```
*****
```

```

if(sezione == ICONS) {

    // titolo
    image(linea_arancione, 0, 46);
    image(icona_quadro, 6, 24);

    textAlign(LEFT);
    textFont(din_black_12);
    text(titolo_quadro[mode], 32, 34);
    textFont(din_light_12);
    text(autore_quadro[mode], 32, 46);

    // testo
    textFont(din_light_15_brown);
    text("Ci sono ", 32, 85);
    // conta indizi
    clues = 0;
    for(i=0;i<=8;i++) {
        if(icons_painting[mode][i] == 1) clues += 1;
    }
    textFont(din_black_15);
    text(""+clues+" ", 88, 85);
    textFont(din_light_15_brown);
    text("indizi qui. Vedi", 102, 85);
    text("qualche ", 32, 102);

    // grafica icone
    image(griglia_icone, 0, 0);

    // SKULL
    if(modeIcon == SKULL) {
        textFont(din_black_15);
        text("TESCHIO", 91, 102);
        textFont(din_light_15_brown);
    }
}

```

```

    text("?", 164, 102);
    // image
    image(select_1_1, 32, 114);
}

// FOOD
if(modeIcon == FOOD) {
    textFont(din_black_15);
    text("CIBO", 91, 102);
    textFont(din_light_15_brown);
    text("?", 134, 102);
    // image
    image(select_1_2, 92, 122);
}

// SHIP
if(modeIcon == SHIP) {
    textFont(din_black_15);
    text("NAVE", 91, 102);
    textFont(din_light_15_brown);
    text("?", 135, 102);
    // image
    image(select_1_3, 150, 122);
}

// WEAPON
if(modeIcon == WEAPON) {
    textFont(din_black_15);
    text("ARMA", 91, 102);
    textFont(din_light_15_brown);
    text("?", 138, 102);
    // image
    image(select_2_1, 32, 182);
}

// JEWEL
if(modeIcon == JEWEL) {
    textFont(din_black_15);
    text("GIOIELLO", 91, 102);
    textFont(din_light_15_brown);
    text("?", 171, 102);
    // image
    image(select_2_2, 92, 182);
}

// PLANT
if(modeIcon == PLANT) {
    textFont(din_black_15);
    text("PIANTA", 91, 102);
    textFont(din_light_15_brown);
    text("?", 153, 102);
    // image
    image(select_2_3, 149, 182);
}

// FLAG
if(modeIcon == FLAG) {
    textFont(din_black_15);
    text("BANDIERA", 91, 102);
    textFont(din_light_15_brown);
    text("?", 175, 102);
    // image
    image(select_3_1, 32, 239);
}

// INSTRUMENT
if(modeIcon == INSTRUMENT) {
    textFont(din_black_15);
    text("STRUMENTO", 91, 102);
    textFont(din_light_15_brown);
}

```

```

    text("?", 190, 102);
    // image
    image(select_3_2, 92, 239);
}

// ANIMAL
if(modeIcon == ANIMAL) {
    textFont(din_black_15);
    text("ANIMALE", 91, 102);
    textFont(din_light_15_brown);
    text("?", 165, 102);
    // image
    image(select_3_3, 149, 239);
}

// icone
if(icons_painting[mode][SKULL] < 2)    image(i_skull, 46, 128);
if(icons_painting[mode][FOOD] < 2)    image(i_food, 100, 129);
if(icons_painting[mode][SHIP] < 2)    image(i_ship, 152, 127);
if(icons_painting[mode][WEAPON] < 2)  image(i_weapon, 37, 185);
if(icons_painting[mode][JEWEL] < 2)   image(i_jewel, 100, 185);
if(icons_painting[mode][PLANT] < 2)   image(i_plant, 155, 186);
if(icons_painting[mode][FLAG] < 2)    image(i_flag, 46, 242);
if(icons_painting[mode][INSTRUMENT] < 2) image(i_instrument, 100, 242);
if(icons_painting[mode][ANIMAL] < 2)  image(i_animal, 153, 242);

drawSoftKeys(true,true,true);

}

// TAKE PHOTOS
*****
if(sezione == TAKE_PHOTO) {

    // show camera
    showCamera(39, 122, 163, 147, imageBackground);

    drawSoftKeys(false,true,true);

    // titolo
    image(linea_arancione, 0, 46);
    image(icona_quadro, 6, 24);

    textAlign(LEFT);
    textFont(din_black_12);
    text(titolo_quadro[mode], 32, 34);
    textFont(din_light_12);
    text(autore_quadro[mode], 32, 46);

    // testo
    textFont(din_light_15_brown);
    text("Take a picture of the ", 32, 85);

    // icone
    if(modeIcon == SKULL) image(i_skull_piccolo, 183, 64);
    if(modeIcon == FOOD) image(i_food_piccolo, 183, 62);
    if(modeIcon == SHIP) image(i_ship_piccolo, 182, 62);
    if(modeIcon == WEAPON) image(i_weapon_piccolo, 182, 64);
    if(modeIcon == JEWEL) image(i_jewel_piccolo, 180, 62);
    if(modeIcon == PLANT) image(i_plant_piccolo, 183, 64);
    if(modeIcon == FLAG) image(i_flag_piccolo, 183, 64);
    if(modeIcon == INSTRUMENT) image(i_instrument_piccolo, 183, 64);
    if(modeIcon == ANIMAL) image(i_animal_piccolo, 183, 62);
}

// GOT CLUES *****
if(sezione == GOT_CLUES) {

```

```

drawSoftKeys(true,true,false);

// titolo
image(linea_arancione, 0, 46);
image(icona_quadro, 6, 24);

textAlign(LEFT);
textFont(din_black_12);
text(titolo_quadro[mode], 32, 34);
textFont(din_light_12);
text(autore_quadro[mode], 32, 46);

// WELL DONE //////////////////////////////////////
println("MODE="+mode);
println("MODEICON="+modeIcon);
println("ICONA="+icons_painting[mode][modeIcon]);

if(icons_painting[mode][modeIcon] == 1) {

    // cornice con l'indizio
    immagineIndizio = mode*10+modeIcon;
    if(immagineIndizio == 2)    image(quadro02, 0,0);
    if(immagineIndizio == 3)    image(quadro03, 0,0);
    if(immagineIndizio == 5)    image(quadro05, 0,0);
    if(immagineIndizio == 6)    image(quadro06, 0,0);
    if(immagineIndizio == 8)    image(quadro08, 0,0);
    if(immagineIndizio == 14)   image(quadro14, 0,0);
    if(immagineIndizio == 15)   image(quadro15, 0,0);
    if(immagineIndizio == 17)   image(quadro17, 0,0);
    if(immagineIndizio == 18)   image(quadro18, 0,0);
    if(immagineIndizio == 23)   image(quadro23, 0,0);
    if(immagineIndizio == 25)   image(quadro25, 0,0);
    if(immagineIndizio == 26)   image(quadro26, 0,0);
    if(immagineIndizio == 28)   image(quadro28, 0,0);
    if(immagineIndizio == 33)   image(quadro33, 0,0);
    if(immagineIndizio == 35)   image(quadro35, 0,0);
    if(immagineIndizio == 37)   image(quadro37, 0,0);
    image(cornice_storta,0,0);

    // spook
    spook =    new Spook("occhei", frameCount, false);
    // text
    image(nuvola_piccola, 12, 75);
    textFont(din_light_15_brown);
    textAlign(LEFT);
    text("Benissimo!!", 32, 104, 165, 40);
    text("Hai collezionato 1", 32, 120, 165, 40);

    // icone
    if(modeIcon == SKULL) image(i_skull_piccolo, 98, 138);
    if(modeIcon == FOOD) image(i_food_piccolo, 98, 138);
    if(modeIcon == SHIP) image(i_ship_piccolo, 98, 138);
    if(modeIcon == WEAPON) image(i_weapon_piccolo, 98, 138);
    if(modeIcon == JEWEL) image(i_jewel_piccolo, 98, 138);
    if(modeIcon == PLANT) image(i_plant_piccolo, 98, 138);
    if(modeIcon == FLAG) image(i_flag_piccolo, 98, 138);
    if(modeIcon == INSTRUMENT) image(i_instrument_piccolo, 98, 138);
    if(modeIcon == ANIMAL) image(i_animal_piccolo, 98, 138);
}

// ALAS!! //////////////////////////////////////
else if(icons_painting[mode][modeIcon] == 0) {

    //neve(48);
    // spook
    spook =    new Spook("piange", frameCount, true);
    // text
    image(nuvola_grande, 12, 75);
    textFont(din_light_15_brown);
    textAlign(LEFT);

```

```

text("Uffa!", 100, 104);
text("Nessun", 32, 143);
text("qui...", 125, 143);
text("Hai perso gli indizi di questo quadro!", 32, 145, 165, 40);

// icone
if(modeIcon == SKULL) image(i_skull_piccolo, 91, 109);
if(modeIcon == FOOD) image(i_food_piccolo, 91, 109);
if(modeIcon == SHIP) image(i_ship_piccolo, 87, 109);
if(modeIcon == WEAPON) image(i_weapon_piccolo, 91, 109);
if(modeIcon == JEWEL) image(i_jewel_piccolo, 90, 109);
if(modeIcon == PLANT) image(i_plant_piccolo, 91, 109);
if(modeIcon == FLAG) image(i_flag_piccolo, 91, 109);
if(modeIcon == INSTRUMENT) image(i_instrument_piccolo, 92, 109);
if(modeIcon == ANIMAL) image(i_animal_piccolo, 88, 109);

// annullo il quadro
paintings_found[mode] =1;
paintings_wrong[mode] =1;
}
}

// ROOM COMPLETE
*****
if(sezione == ROOM_COMPLETE) {

drawSoftKeys(true,true,false);

// disegno stanzina di sfondo
image(room_complete, 0, 93);

// coriandoli
coriandoli(50, "discesa");

// titolo
image(linea_arancione, 0, 46);
image(icona_stanza, 6, 24);
textAlign(LEFT);
textFont(din_black_12);
text("STANZA 14", 32, 34);

// spook
spook = new Spook("esulta", frameCount, false);
// text
image(nuvola_piccola, 12, 75);
textFont(din_light_15_brown);
textAlign(LEFT);
text("Ok, "+nickname+", hai finito questa stanza! Corri alla prossima!", 32, 104, 165, 70);
}

// STAND_BY *****
if(sezione == STAND_BY_TO_EXIT) {

background(0); // black background
// text
textFont(din_light_15_white);
textAlign(LEFT);
text("L'applicazione va in stand-by finchè il bambino non entra in un'altra stanza o esce dal
museo.", 32, 127, 175, 150);
}

// EXIT MUSEUM *****
if(sezione == EXIT_MUSEUM) {

// spook
spook = new Spook("giravolta", frameCount, true);
// text
image(nuvola_grande, 12, 75);
}

```

```

    textFont(din_light_15_brown);
    textAlign(LEFT);
    text("Ce l'abbiamo fatta!", 32, 120);
    text("Ora possiamo usare gli indizi raccolti per risolvere il mistero di Hao Dong! Iniziamo?",
32, 128, 165, 100);

    // titolo
    image(linea_arancione, 0, 46);
    image(icona_spook, 6, 24);
    textAlign(LEFT);
    textFont(din_black_12);
    text("INVESTIGAZIONE COMPLETA!", 32, 34);

    drawSoftKeys(true,true,false);
}

// FINAL QUESTION
*****
if(sezione == FINAL_QUESTION) {

    // scrolling bar
    strokeWeight(4);
    //// sfondo
    stroke(dark_yellow);
    line(232,82,232,210);
    //// bar
    stroke(orange);
    if(mode > QUESTION_1) { // barra + corta
        pushMatrix();
        translate(0, 24*scrollingText);
        line(232,82+24*mode,232,210); // lunghezza barra = tolgo 1/5 di barra per ogni
parte di quiz (5 in tutto) caricata
        popMatrix();
    }
    else line(232,82,232,210);

    // final question test //////////////////////////////////////

    pushMatrix(); // in questo modo (matrix + translate) posso spostare TUTTI i
layer di testo nella matrix
    if(modeXbarra==QUESTION_1) {
        modeYquestion = easeTo(modeYquestion, POSITION_QUESTION_1, easeSpeed);
    }
    if(modeXbarra == QUESTION_2) {
        modeYquestion = easeTo(modeYquestion, POSITION_QUESTION_2, easeSpeed);
    }
    else if(modeXbarra == QUESTION_3) {
        modeYquestion = easeTo(modeYquestion, POSITION_QUESTION_3, easeSpeed);
    }
    else if(modeXbarra == QUESTION_4) {
        modeYquestion = easeTo(modeYquestion, POSITION_QUESTION_4, easeSpeed);
    }
    else if(modeXbarra == QUESTION_5) {
        modeYquestion = easeTo(modeYquestion, POSITION_QUESTION_5, easeSpeed);
    }
    else if(modeXbarra == QUESTION_6) {
        modeYquestion = easeTo(modeYquestion, POSITION_QUESTION_6, easeSpeed);
    }
    translate(0, modeYquestion);

    textFont(din_light_15_brown);
    textAlign(LEFT);
    text("Come pensavamo,", 32, 72);
    text("Marco Polo non aveva", 32, 96);
    text("lasciato Hao Dong... ", 32, 120);
    if(mode > QUESTION_1) {
        text("Era salpato verso l'Asia", 32, 144);
        text("per recuperare, da", 32, 168);
    }
}

```

```

text("Qubilay Khan, il", 32, 192);
text("bellissimo", 32, 216);
emptyImage = new emptyImage(147-37, 216-19);
}
if(mode > QUESTION_2) {
  if(STARTING_QUESTION_2 < 10) STARTING_QUESTION_2 += 1;
  selectPicture(147-37, 216-19, given_answers[QUESTION_2], STARTING_QUESTION_2);
  text("a cui lei", 153, 216);
  text("era affezionata sin da", 32, 240);
  text("bambina. Durante il", 32, 264);
  text("viaggio di ritorno, la", 32, 288);
  emptyImage = new emptyImage(174, 288-19);
}
if(mode > QUESTION_3) {
  if(STARTING_QUESTION_3 < 10) STARTING_QUESTION_3 += 1;
  selectPicture(174, 288-19, given_answers[QUESTION_3], STARTING_QUESTION_3);
  text("di Marco, sventolando la", 32, 312);
  emptyImage = new emptyImage(32, 336-19);
}
if(mode > QUESTION_4) {
  if(STARTING_QUESTION_4 < 10) STARTING_QUESTION_4 += 1;
  selectPicture(32, 336-19, given_answers[QUESTION_4], STARTING_QUESTION_4);
  text("veneziana, venne", 73, 336);
  text("attaccata dall'esercito", 32, 360);
  text("Turco, famoso per le sue", 32, 382);
  emptyImage = new emptyImage(32, 406-19);
}
if(mode > QUESTION_5) {
  if(STARTING_QUESTION_5 < 10) STARTING_QUESTION_5 += 1;
  selectPicture(32, 406-19, given_answers[QUESTION_5], STARTING_QUESTION_5);
  text("letali. Marco", 73, 406);
  text("fu catturato e passò", 32, 430);
  text("il resto della sua vita", 32, 454);
  text("pensando solo a lei.", 32, 478);
}
popMatrix();

// mask scrolling text
image(final_mask_testo, 0, 0); // nell'ultima domanda non devo mascherare il testo in
basso!

// ripristino la upper bar
textAlign(LEFT);
textFont(din_light_12_light_brown);
text("Spook", 6, 13); // application title
textAlign(RIGHT);
text(totPunti+" pt", 234, 13); // points

// the end (se sono all'ultimo quesito
if(mode == QUESTION_6) {
  textFont(din_black_15_brown);
  textAlign(CENTER);
  text("Fine", 120, 252);
}

// titolo
image(linea_arancione, 0, 46);
image(icona_spook, 6, 24);
textAlign(LEFT);
textFont(din_black_12);
text("RISOLVIAMO IL MISTERO", 32, 34);
textFont(din_light_12);
text("Completa la storia con gli indizi", 32, 46);

// disegno la barra delle icone ////////////////
// solo se devo selezionarne una XD (quindi non in QUESTION_1 e 6)
if(mode > QUESTION_1 && mode < QUESTION_6) {

```



```

// selezionatore
image(final_selection, 85, 217);

// icone
// SKULL ///
if(modeIcon == SKULL) {
    // image
    modeXicon = easeTo(modeXicon, POSITION_SKULL, easeSpeed);
    image(final_icone, modeXicon, 232);
}

// FOOD ///
if(modeIcon == FOOD) {
    // image
    modeXicon = easeTo(modeXicon, POSITION_FOOD, easeSpeed);
    image(final_icone, modeXicon, 232);
}

// SHIP ///
if(modeIcon == SHIP) {
    // image
    modeXicon = easeTo(modeXicon, POSITION_SHIP, easeSpeed);
    image(final_icone, modeXicon, 232);
}

// WEAPON ///
if(modeIcon == WEAPON) {
    // image
    modeXicon = easeTo(modeXicon, POSITION_WEAPON, easeSpeed);
    image(final_icone, modeXicon, 232);
}

// JEWEL ///
if(modeIcon == JEWEL) {
    // image
    modeXicon = easeTo(modeXicon, POSITION_JEWEL, easeSpeed);
    image(final_icone, modeXicon, 232);
}

// PLANT ///
if(modeIcon == PLANT) {
    // image
    modeXicon = easeTo(modeXicon, POSITION_PLANT, easeSpeed);
    image(final_icone, modeXicon, 232);
}

// FLAG ///
if(modeIcon == FLAG) {
    // image
    modeXicon = easeTo(modeXicon, POSITION_FLAG, easeSpeed);
    image(final_icone, modeXicon, 232);
}

// INSTRUMENT ///
if(modeIcon == INSTRUMENT) {
    // image
    modeXicon = easeTo(modeXicon, POSITION_INSTRUMENT, easeSpeed);
    image(final_icone, modeXicon, 232);
}

// ANIMAL ///
if(modeIcon == ANIMAL) {
    // image
    modeXicon = easeTo(modeXicon, POSITION_ANIMAL, easeSpeed);
    image(final_icone, modeXicon, 232);
}

// finta maschera per le icone

```

```

        image(final_mask, 0, 227);

        // freccine
        if(modeIcon < 8) image(frecce_dx, 220, 246);
        if(modeIcon > 0) image(frecce_sx, 5, 246);
    }
    drawSoftKeys(true,true,false);
}

// LAST SCREEN
*****
if(sezione == LAST_SCREEN) {

    // coriandoli
    coriandoli(100, "salita");

    // titolo
    image(linea_arancione, 0, 46);
    image(icona_spook, 6, 24);
    textAlign(LEFT);
    textFont(din_black_12);
    text("CONGRATULAZIONI!!", 32, 34);

    if(mode==CASE_SOLVED) {
        drawSoftKeys(false,true,false);
        moodSpook = "giravolta";
        image(final_screen, 0, 0);
        // text
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("Congratulazioni, "+nickname+"! Il mistero è stato risolto e Hao Dong ora può riposare
in pace!", 32, 72, 190, 200);
    }

    if(mode==GAME_COMPLETE) {
        moodSpook = "saluta";
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("Hai completato il gioco con", 32, 72, 190, 200);
        textFont(din_black_15_brown);
        textAlign(CENTER);
        text(totPunti+" punti", 120, 119);
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("E' stato un vero piacere giocare al detective con te!", 32, 130, 190, 200);
        text("Ci vediamo!!", 32, 230);
    }

    // spook
    spook = new Spook(moodSpook, frameCount, false);
}

// *****
// ** TIME CHALLENGE
// *****

void drawTimeChallenge (int sezione, int mode) // sezione, sottosezione
{

```

```

// NARRATIVE *****
if(sezione == NARRATIVE) {

    image(fumini,0,0);

    // IN_THIS_MUSEUM //////////////////////////////////////
    if(mode == IN_THIS_MUSEUM) {
        moodSpook = "saluta";           // changes spook's mood in the spook animation

        // text
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("In questo museo ci sono degli indizi sullo spirito di Hao Dong, figlia di Qubilay Khan
e moglie di Marco Polo.", 32, 72, 180, 200);

        // image slide: museum
        X_story_2 = easeTo(X_story_2, POSITION_MUSEUM, easeSpeed);
        image(story_02, X_story_2, 0);
    }

    // SIGNORA MARCO POLO //////////////////////////////////////
    if(mode == MARCO_POLO_LEFT) {
        moodSpook = "triste";

        // text
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("Le hanno fatto credere che Marco Polo l'abbia abbandonata per sempre. Lei non riesce a
farsene una ragione... Ma sono sicuro che c'è qualcosa sotto!", 32, 72, 180, 200);

        // barchetta che passa
        if(Xbarchetta <= 300) Xbarchetta = Xbarchetta + 1;
        image(story_03, Xbarchetta, 208);

        // image slide: signora marco polo
        X_story_2 = easeTo(X_story_2, POSITION_KHAN, easeSpeed);
        image(story_02, X_story_2, 0);
    }

    // RECOGNIZE_CLUES //////////////////////////////////////
    if(mode == RECOGNIZE_CLUES) {
        moodSpook = "felice";

        // image slide: mobile phone
        //X_story_2 = easeTo(X_story_2, POSITION_PHONE, easeSpeed);
        //image(story_02, X_story_2, 0);

        // text
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("Trova gli indizi nascosti nei quadri del museo.", 32, 72, 180, 150);
    }

    // AT_THE_END //////////////////////////////////////
    if(mode == AT_THE_END) {
        moodSpook = "felice";

        // image slide
        //X_story_2 = easeTo(X_story_2, -700, easeSpeed);
        //image(story_02, X_story_2, 0);

        // text
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("Trova gli indizi nascosti nei quadri del museo.", 32, 72, 180, 150);
        text("Alla fine cercheremo di ricostruire la storia della povera Hao Dong.", 32, 127, 175,
150);
    }
}

```



```

moodSpook = "felice";

// text
textFont(din_light_15_brown);
textAlign(LEFT);
text("Remember not to get the wrong clues otherwise we will not be able to use that painting
as evidence.", 32, 72, 180, 150);
text("Are you ready?.", 32, 155, 170, 150);
text("Enter the first room to start playing.", 32, 177, 120, 150);
}

// drawing spook
spook = new Spook(moodSpook, frameCount, false);

// titolo
image(linea_arancione, 0, 46);
textFont(din_black_12);
textAlign(LEFT);
text("INTRODUZIONE", 32, 34);
image(icona_spook, 6, 24);

drawSoftKeys(false,true,false);

// STAND_BY //////////////////////////////////////
if(mode == STAND_BY) {

    background(0); // black background
    // text
    textFont(din_light_15_white);
    textAlign(LEFT);
    text("L'applicazione va in stand-by finchè il bambino non entra in una stanza.", 32, 147,
175, 150);
}

}

// MUSEUM_ROOM
*****
if(sezione == MUSEUM_ROOM) {
    timeCheck();
    drawSoftKeys(true,true,false);

    // titolo
    image(linea_arancione, 0, 46);
    drawCursor();
    textFont(din_black_12);
    textAlign(LEFT);
    text("ROOM 14", 32, 34);
    image(icona_stanza, 6, 24);

    // testo
    textFont(din_light_15_brown);
    text("Guardati in giro e trova uno di questi quadri...", 32, 72, 190, 70);

    // paintings list
    // 1 //////////////////////////////////////
    if(mode==PAINT_1) {
        // selezione
        image(selezione_grande_1, 3, 106);
        textFont(din_black_12_white);
    }
    else {
        textFont(din_black_12_brown);
    }
    if(paintings_found[PAINT_1] == 0) {
        image(icona_quadro, 6, 117);
    }
}

```

```

text("L'ARRIVO DEI PELLEGRINI", 32, 128);

if(mode==PAINT_1) {
    textFont(din_light_12_white);
}
else {
    textFont(din_light_12_brown);
}
text("Carpaccio (1490)", 32, 140);

// 2 //////////////////////////////////////
if(mode==PAINT_2) {
    image(selezione_grande_2, 3, 153);
    textFont(din_black_12_white);
}
else {
    textFont(din_black_12_brown);
}
if(paintings_found[PAINT_2] == 0) {
    image(icona_quadro, 6, 162);
}
text("SANTA VENERANDA IN TRONO", 32, 173);
if(mode==PAINT_2) {
    textFont(din_light_12_white);
}
else {
    textFont(din_light_12_brown);
}
text("Bastiani (1470)", 32, 185);

// 3 //////////////////////////////////////
if(mode==PAINT_3) {
    image(selezione_grande_1, 3, 196);
    textFont(din_black_12_white);
}
else {
    textFont(din_black_12_brown);
}
if(paintings_found[PAINT_3] == 0) {
    image(icona_quadro, 6, 207);
}
text("APOTEOSI DEI DIECIMILA MARTIRI", 32, 218);
if(mode==PAINT_3) {
    textFont(din_light_12_white);
}
else {
    textFont(din_light_12_brown);
}
text("Carpaccio (1515)", 32, 230);

// 4 //////////////////////////////////////
if(mode==PAINT_4) {
    image(selezione_grande_2, 3, 242);
    textFont(din_black_12_white);
}
else {
    textFont(din_black_12_brown);
}
if(paintings_found[PAINT_4] == 0) {
    image(icona_quadro, 6, 252);
}
text("PALA DI SAN GIOBBE", 32, 263);
if(mode==PAINT_4) {
    textFont(din_light_12_white);
}
else {
    textFont(din_light_12_brown);
}
text("Bellini (1487)", 32, 275);
}

```

```

// ICONS SCREEN
*****
if(sezione == ICONS) {

    timeCheck();
    // titolo
    image(linea_arancione, 0, 46);
    image(icona_quadro, 6, 24);

    textAlign(LEFT);
    textFont(din_black_12);
    text(titolo_quadro[mode], 32, 34);
    textFont(din_light_12);
    text(autore_quadro[mode], 32, 46);

    // testo
    textFont(din_light_15_brown);
    text("Ci sono ", 32, 85);
    // conta indizi
    clues = 0;
    for(i=0;i<=8;i++) {
        if(icons_painting[mode][i] == 1) clues += 1;
    }
    textFont(din_black_15);
    text(""+clues+" ", 88, 85);
    textFont(din_light_15_brown);
    text("indizi qui. Vedi", 102, 85);
    text("qualche ", 32, 102);

    // grafica icone
    image(griglia_icone, 0, 0);

    // SKULL
    if(modeIcon == SKULL) {
        textFont(din_black_15);
        text("TESCHIO", 91, 102);
        textFont(din_light_15_brown);
        text("?", 164, 102);
        // image
        image(select_1_1, 32, 114);
    }

    // FOOD
    if(modeIcon == FOOD) {
        textFont(din_black_15);
        text("CIBO", 91, 102);
        textFont(din_light_15_brown);
        text("?", 134, 102);
        // image
        image(select_1_2, 92, 122);
    }

    // SHIP
    if(modeIcon == SHIP) {
        textFont(din_black_15);
        text("NAVE", 91, 102);
        textFont(din_light_15_brown);
        text("?", 135, 102);
        // image
        image(select_1_3, 150, 122);
    }

    // WEAPON
    if(modeIcon == WEAPON) {
        textFont(din_black_15);
        text("ARMA", 91, 102);
        textFont(din_light_15_brown);

```

```

    text("?", 138, 102);
    // image
    image(select_2_1, 32, 182);
}

// JEWEL
if(modeIcon == JEWEL) {
    textFont(din_black_15);
    text("GIOIELLO", 91, 102);
    textFont(din_light_15_brown);
    text("?", 171, 102);
    // image
    image(select_2_2, 92, 182);
}

// PLANT
if(modeIcon == PLANT) {
    textFont(din_black_15);
    text("PIANTA", 91, 102);
    textFont(din_light_15_brown);
    text("?", 153, 102);
    // image
    image(select_2_3, 149, 182);
}

// FLAG
if(modeIcon == FLAG) {
    textFont(din_black_15);
    text("BANDIERA", 91, 102);
    textFont(din_light_15_brown);
    text("?", 175, 102);
    // image
    image(select_3_1, 32, 239);
}

// INSTRUMENT
if(modeIcon == INSTRUMENT) {
    textFont(din_black_15);
    text("STRUMENTO", 91, 102);
    textFont(din_light_15_brown);
    text("?", 190, 102);
    // image
    image(select_3_2, 92, 239);
}

// ANIMAL
if(modeIcon == ANIMAL) {
    textFont(din_black_15);
    text("ANIMALE", 91, 102);
    textFont(din_light_15_brown);
    text("?", 165, 102);
    // image
    image(select_3_3, 149, 239);
}

// icone
if(icons_painting[mode][SKULL] < 2)    image(i_skull, 46, 128);
if(icons_painting[mode][FOOD] < 2)    image(i_food, 100, 129);
if(icons_painting[mode][SHIP] < 2)    image(i_ship, 152, 127);
if(icons_painting[mode][WEAPON] < 2)  image(i_weapon, 37, 185);
if(icons_painting[mode][JEWEL] < 2)  image(i_jewel, 100, 185);
if(icons_painting[mode][PLANT] < 2)  image(i_plant, 155, 186);
if(icons_painting[mode][FLAG] < 2)   image(i_flag, 46, 242);
if(icons_painting[mode][INSTRUMENT] < 2) image(i_instrument, 100, 242);
if(icons_painting[mode][ANIMAL] < 2)  image(i_animal, 153, 242);

drawSoftKeys(true,true,true);
}

```



```

// TAKE PHOTOS
*****
if(sezione == TAKE_PHOTO) {

    timeCheck();
    // show camera
    showCamera(39, 122, 163, 147, imageBackground);

    drawSoftKeys(false,true,true);

    // titolo
    image(linea_arancione, 0, 46);
    drawCursor();
    image(icona_quadro, 6, 24);

    textAlign(LEFT);
    textFont(din_black_12);
    text(titolo_quadro[mode], 32, 34);
    textFont(din_light_12);
    text(autore_quadro[mode], 32, 46);

    // testo
    textFont(din_light_15_brown);
    text("Take a picture of the ", 32, 85);

    // icone
    if(modeIcon == SKULL) image(i_skull_piccolo, 183, 64);
    if(modeIcon == FOOD) image(i_food_piccolo, 183, 62);
    if(modeIcon == SHIP) image(i_ship_piccolo, 182, 62);
    if(modeIcon == WEAPON) image(i_weapon_piccolo, 182, 64);
    if(modeIcon == JEWEL) image(i_jewel_piccolo, 180, 62);
    if(modeIcon == PLANT) image(i_plant_piccolo, 183, 64);
    if(modeIcon == FLAG) image(i_flag_piccolo, 183, 64);
    if(modeIcon == INSTRUMENT) image(i_instrument_piccolo, 183, 64);
    if(modeIcon == ANIMAL) image(i_animal_piccolo, 183, 62);
}

// GOT CLUES *****
if(sezione == GOT_CLUES) {

    timeCheck();
    drawSoftKeys(true,true,false);

    // titolo
    image(linea_arancione, 0, 46);
    drawCursor();
    image(icona_quadro, 6, 24);

    textAlign(LEFT);
    textFont(din_black_12);
    text(titolo_quadro[mode], 32, 34);
    textFont(din_light_12);
    text(autore_quadro[mode], 32, 46);

    // WELL DONE ///////////////////////////////////
    println("MODE="+mode);
    println("MODEICON="+modeIcon);
    println("ICONA="+icons_painting[mode][modeIcon]);

    if(icons_painting[mode][modeIcon] == 1) {

        // cornice con l'indizio
        immagineIndizio = mode*10+modeIcon;
        if(immagineIndizio == 2) image(quadro02, 0,0);
        if(immagineIndizio == 3) image(quadro03, 0,0);
        if(immagineIndizio == 5) image(quadro05, 0,0);
        if(immagineIndizio == 6) image(quadro06, 0,0);
        if(immagineIndizio == 8) image(quadro08, 0,0);
    }
}

```

```

if(immagineIndizio == 14) image(quadro14, 0,0);
if(immagineIndizio == 15) image(quadro15, 0,0);
if(immagineIndizio == 17) image(quadro17, 0,0);
if(immagineIndizio == 18) image(quadro18, 0,0);
if(immagineIndizio == 23) image(quadro23, 0,0);
if(immagineIndizio == 25) image(quadro25, 0,0);
if(immagineIndizio == 26) image(quadro26, 0,0);
if(immagineIndizio == 28) image(quadro28, 0,0);
if(immagineIndizio == 33) image(quadro33, 0,0);
if(immagineIndizio == 35) image(quadro35, 0,0);
if(immagineIndizio == 37) image(quadro37, 0,0);
image(cornice_storta,0,0);

// spook
spook = new Spook("occhei", frameCount, false);
// text
image(nuvola_piccola, 12, 75);
textFont(din_light_15_brown);
textAlign(LEFT);
text("Benissimo!!", 32, 104, 165, 40);
text("Hai collezionato 1", 32, 120, 165, 40);

// icone
if(modeIcon == SKULL) image(i_skull_piccolo, 98, 138);
if(modeIcon == FOOD) image(i_food_piccolo, 98, 138);
if(modeIcon == SHIP) image(i_ship_piccolo, 98, 138);
if(modeIcon == WEAPON) image(i_weapon_piccolo, 98, 138);
if(modeIcon == JEWEL) image(i_jewel_piccolo, 98, 138);
if(modeIcon == PLANT) image(i_plant_piccolo, 98, 138);
if(modeIcon == FLAG) image(i_flag_piccolo, 98, 138);
if(modeIcon == INSTRUMENT) image(i_instrument_piccolo, 98, 138);
if(modeIcon == ANIMAL) image(i_animal_piccolo, 98, 138);
}

// ALAS!! //////////////////////////////////////
else if(icons_painting[mode][modeIcon] == 0) {

//neve(48);
// spook
spook = new Spook("piange", frameCount, true);
// text
image(nuvola_grande, 12, 75);
textFont(din_light_15_brown);
textAlign(LEFT);
text("Uffa!", 100, 104);
text("Nessun", 32, 143);
text("qui...", 125, 143);
text("Hai perso gli indizi di questo quadro!", 32, 145, 165, 40);

// icone
if(modeIcon == SKULL) image(i_skull_piccolo, 91, 109);
if(modeIcon == FOOD) image(i_food_piccolo, 91, 109);
if(modeIcon == SHIP) image(i_ship_piccolo, 87, 109);
if(modeIcon == WEAPON) image(i_weapon_piccolo, 91, 109);
if(modeIcon == JEWEL) image(i_jewel_piccolo, 90, 109);
if(modeIcon == PLANT) image(i_plant_piccolo, 91, 109);
if(modeIcon == FLAG) image(i_flag_piccolo, 91, 109);
if(modeIcon == INSTRUMENT) image(i_instrument_piccolo, 92, 109);
if(modeIcon == ANIMAL) image(i_animal_piccolo, 88, 109);

// annullo il quadro
paintings_found[mode] =1;
paintings_wrong[mode] =1;
}
}

// ROOM COMPLETE
*****
if(sezione == ROOM_COMPLETE) {
if(endTime == 0){

```

```

    endTime = millis();
    for (i=0; i<=3; i++){
        if(paintings_wrong[i] == 1){
            sbaglio ++;
        }
    }
    totPunti += ((360000 - (endTime - starTime)) / 1000)-(90*sbaglio);
}
drawSoftKeys(true,true,false);

// disegno stanzina di sfondo
image(room_complete, 0, 93);

// coriandoli
coriandoli(50, "discesa");

// titolo
image(linea_arancione, 0, 46);
image(icona_stanza, 6, 24);
textAlign(LEFT);
textFont(din_black_12);
text("14", 32, 34);

// spook
spook = new Spook("esulta", frameCount, true);
// text
image(nuvola_piccola, 12, 75);
textFont(din_light_15_brown);
textAlign(LEFT);
text("Ok, "+nickname+", hai finito questa stanza! Corri alla prossima!", 32, 104, 165, 70);
}

// TIME IS UP *****
if(sezione == TIME_IS_UP) {

    drawSoftKeys(false,true,false);

    // titolo
    image(linea_arancione, 0, 46);
    image(icona_stanza, 6, 24);
    textAlign(LEFT);
    textFont(din_black_12);
    text("14", 32, 34);

    // spook
    spook = new Spook("paura", frameCount, true);
    // text
    image(nuvola_piccola, 12, 75);
    textFont(din_light_15_brown);
    textAlign(LEFT);
    text("No! "+nickname+", è finito il tempo! Corri nella prossima stanza.", 32, 104, 155, 70);
}

// STAND_BY *****
if(sezione == STAND_BY_TO_EXIT) {

    background(0); // black background
    // text
    textFont(din_light_15_white);
    textAlign(LEFT);
    text("L'applicazione va in stand-by finchè il bambino non entra in un'altra stanza o esce dal
museo.", 32, 127, 175, 150);
}

// EXIT MUSEUM *****
if(sezione == EXIT_MUSEUM) {

    // spook

```

```

spook = new Spook("giravolta", frameCount, true);
// text
image(nuvola_grande, 12, 75);
textFont(din_light_15_brown);
textAlign(LEFT);
text("Ce l'abbiamo fatta!", 32, 120);
text("Ora possiamo usare gli indizi raccolti per risolvere il mistero di Hao Dong! Iniziamo?",
32, 128, 165, 100);

// titolo
image(linea_arancione, 0, 46);
image(icona_spook, 6, 24);
textAlign(LEFT);
textFont(din_black_12);
text("INVESTIGAZIONE COMPLETA!", 32, 34);

drawSoftKeys(true,true,false);
}

// FINAL QUESTION
*****
if(sezione == FINAL_QUESTION) {

// scrolling bar
strokeWeight(4);
//// sfondo
stroke(dark_yellow);
line(232,82,232,210);
//// bar
stroke(orange);
if(mode > QUESTION_1) { // barra + corta
pushMatrix();
translate(0, 24*scrollingText);
line(232,82+24*mode,232,210); // lunghezza barra = tolgo 1/5 di barra per ogni
parte di quiz (5 in tutto) caricata
popMatrix();
}
else line(232,82,232,210);

// final question test //////////////////////////////////////

pushMatrix(); // in questo modo (matrix + translate) posso spostare TUTTI i
layer di testo nella matrix
if(modeXbarra==QUESTION_1) {
modeYquestion = easeTo(modeYquestion, POSITION_QUESTION_1, easeSpeed);
}
if(modeXbarra == QUESTION_2) {
modeYquestion = easeTo(modeYquestion, POSITION_QUESTION_2, easeSpeed);
}
else if(modeXbarra == QUESTION_3) {
modeYquestion = easeTo(modeYquestion, POSITION_QUESTION_3, easeSpeed);
}
else if(modeXbarra == QUESTION_4) {
modeYquestion = easeTo(modeYquestion, POSITION_QUESTION_4, easeSpeed);
}
else if(modeXbarra == QUESTION_5) {
modeYquestion = easeTo(modeYquestion, POSITION_QUESTION_5, easeSpeed);
}
else if(modeXbarra == QUESTION_6) {
modeYquestion = easeTo(modeYquestion, POSITION_QUESTION_6, easeSpeed);
}
translate(0, modeYquestion);

textFont(din_light_15_brown);
textAlign(LEFT);
text("Come pensavamo,", 32, 72);
text("Marco Polo non aveva", 32, 96);
text("lasciato Hao Dong... ", 32, 120);

```

```

if(mode > QUESTION_1) {
    text("Era salpato verso l'Asia", 32, 144);
    text("per recuperare, da", 32, 168);
    text("Qubilay Khan, il", 32, 192);
    text("bellissimo", 32, 216);
    emptyImage = new emptyImage(147-37, 216-19);
}
if(mode > QUESTION_2) {
    if(STARTING_QUESTION_2 < 10) STARTING_QUESTION_2 += 1;
    selectPicture(147-37, 216-19, given_answers[QUESTION_2], STARTING_QUESTION_2);
    text("a cui lei", 153, 216);
    text("era affezionata sin da", 32, 240);
    text("bambina. Durante il", 32, 264);
    text("viaggio di ritorno, la", 32, 288);
    emptyImage = new emptyImage(174, 288-19);
}
if(mode > QUESTION_3) {
    if(STARTING_QUESTION_3 < 10) STARTING_QUESTION_3 += 1;
    selectPicture(174, 288-19, given_answers[QUESTION_3], STARTING_QUESTION_3);
    text("di Marco, sventolando la", 32, 312);
    emptyImage = new emptyImage(32, 336-19);
}
if(mode > QUESTION_4) {
    if(STARTING_QUESTION_4 < 10) STARTING_QUESTION_4 += 1;
    selectPicture(32, 336-19, given_answers[QUESTION_4], STARTING_QUESTION_4);
    text("veneziana, venne", 73, 336);
    text("attaccata dall'esercito", 32, 360);
    text("Turco, famoso per le sue", 32, 382);
    emptyImage = new emptyImage(32, 406-19);
}
if(mode > QUESTION_5) {
    if(STARTING_QUESTION_5 < 10) STARTING_QUESTION_5 += 1;
    selectPicture(32, 406-19, given_answers[QUESTION_5], STARTING_QUESTION_5);
    text("letali. Marco", 73, 406);
    text("fu catturato e passò", 32, 430);
    text("il resto della sua vita", 32, 454);
    text("pensando solo a lei.", 32, 478);
}
popMatrix();

// mask scrolling text
image(final_mask_testo, 0, 0); // nell'ultima domanda non devo mascherare il testo in
basso!

// ripristino la upper bar
textAlign(LEFT);
textFont(din_light_12_light_brown);
text("Spook", 6, 13); // application title
textAlign(RIGHT);
text(totPunti+" pt", 234, 13); // points

// the end (se sono all'ultimo quesito
if(mode == QUESTION_6) {
    textFont(din_black_15_brown);
    textAlign(CENTER);
    text("Fine", 120, 252);
}

// titolo
image(linea_arancione, 0, 46);
image(icona_spook, 6, 24);
textAlign(LEFT);
textFont(din_black_12);
text("RISOLVIAMO IL MISTERO", 32, 34);
textFont(din_light_12);
text("Completa la storia con gli indizi", 32, 46);

```

```

// disegno la barra delle icone //////////////////////////////////
// solo se devo selezionarne una XD (quindi non in QUESTION_1 e 6)
if(mode > QUESTION_1 && mode < QUESTION_6) {

    // selezionatore
    image(final_selection, 85, 217);

    // icone
    // SKULL ///
    if(modeIcon == SKULL) {
        // image
        modeXicon = easeTo(modeXicon, POSITION_SKULL, easeSpeed);
        image(final_icone, modeXicon, 232);
    }

    // FOOD ///
    if(modeIcon == FOOD) {
        // image
        modeXicon = easeTo(modeXicon, POSITION_FOOD, easeSpeed);
        image(final_icone, modeXicon, 232);
    }

    // SHIP ///
    if(modeIcon == SHIP) {
        // image
        modeXicon = easeTo(modeXicon, POSITION_SHIP, easeSpeed);
        image(final_icone, modeXicon, 232);
    }

    // WEAPON ///
    if(modeIcon == WEAPON) {
        // image
        modeXicon = easeTo(modeXicon, POSITION_WEAPON, easeSpeed);
        image(final_icone, modeXicon, 232);
    }

    // JEWEL ///
    if(modeIcon == JEWEL) {
        // image
        modeXicon = easeTo(modeXicon, POSITION_JEWEL, easeSpeed);
        image(final_icone, modeXicon, 232);
    }

    // PLANT ///
    if(modeIcon == PLANT) {
        // image
        modeXicon = easeTo(modeXicon, POSITION_PLANT, easeSpeed);
        image(final_icone, modeXicon, 232);
    }

    // FLAG ///
    if(modeIcon == FLAG) {
        // image
        modeXicon = easeTo(modeXicon, POSITION_FLAG, easeSpeed);
        image(final_icone, modeXicon, 232);
    }

    // INSTRUMENT ///
    if(modeIcon == INSTRUMENT) {
        // image
        modeXicon = easeTo(modeXicon, POSITION_INSTRUMENT, easeSpeed);
        image(final_icone, modeXicon, 232);
    }

    // ANIMAL ///
    if(modeIcon == ANIMAL) {
        // image
        modeXicon = easeTo(modeXicon, POSITION_ANIMAL, easeSpeed);
        image(final_icone, modeXicon, 232);
    }
}

```

```

    }

    // finta maschera per le icone
    image(final_mask, 0, 227);

    // frecce
    if(modeIcon < 8) image(frecce_dx, 220, 246);
    if(modeIcon > 0) image(frecce_sx, 5, 246);

}
drawSoftKeys(true,true,false);
}

// LAST SCREEN
*****
if(sezione == LAST_SCREEN) {

    // coriandoli
    coriandoli(100, "salita");

    // titolo
    image(linea_arancione, 0, 46);
    image(icona_spook, 6, 24);
    textAlign(LEFT);
    textFont(din_black_12);
    text("CONGRATULAZIONI!!", 32, 34);

    if(mode==CASE_SOLVED) {
        drawSoftKeys(false,true,false);
        moodSpook = "giravolta";
        image(final_screen, 0, 0);
        // text
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("Congratulazioni, "+nickname+"! Il mistero è stato risolto e Hao Dong ora può riposare
in pace!", 32, 72, 190, 200);
    }

    if(mode==GAME_COMPLETE) {
        moodSpook = "saluta";
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("Hai completato il gioco con", 32, 72, 190, 200);
        textFont(din_black_15_brown);
        textAlign(CENTER);
        text(totPunti+" punti", 120, 119);
        textFont(din_light_15_brown);
        textAlign(LEFT);
        text("E' stato un vero piacere giocare al detective con te!", 32, 130, 190, 200);
        text("Ci vediamo!!", 32, 230);
    }

    // spook
    spook = new Spook(moodSpook, frameCount, false);
}
}

/*****
* LOGIC SECTION *****/
*****/

import java.lang.Math;
import processing.video.*;
import processing.image2.*;

```

```

import processing.phone.*; // import phone library to go fullscreen
Phone myPhone;           // named reference to phone instance
Capture myCapture;       // to use a capture object

////////////////////////////////////
//  SETUP  //////////////////////////////////////
////////////////////////////////////

void setup() {
  // go fullscreen
  myPhone = new Phone(this); // create new phone instance/controller
  myPhone.fullscreen();      // tell phone to go fullscreen

  myCapture = new Capture(this); //for using the camera

  framerate(30);

  // COLORS //////////////////////////////////
  brown = color(132,83,42);
  orange = color(246,146,30);
  white = color(255,255,255);
  yellow = color(255,255,210);
  black = color(0,0,0);
  light_brown = color(188,141,51);
  dark_yellow = color(255,228,0);

  // FUNZIONI //////////////////////////////////
  loadImages();
  loadFonts();
  multitap(); // tastiera (numerica del cell) attiva per scrivere il nickname

  // CREO I PALLINI
  for(int p=0;p<=100;p++) {
    //create random coords
    int randomX = random(0, width);
    int randomY = random(330, 350);
    int randomW = random(1,8);
    int randomT = random(20,255);
    //randomX = constrain(randomX,0+randomW,width);
    //randomY = constrain(randomY,0,height-randomW);

    // store data
    pallinoX = append(pallinoX, randomX);
    pallinoY = append(pallinoY, randomY);
    pallinoW = append(pallinoW, randomW);
    pallinoT = append(pallinoT, randomT);
    pallinoTime = append(pallinoTime, random(0,10));
    pallinoDirection = append(pallinoDirection, -1);
    pallinoSpeed = append(pallinoSpeed, random(5,10));
    pallinoColor = append(pallinoColor, 100);
  }

  /*
  // CREO LA NEVE
  for(int p=0;p<=50;p++) {
    //create random coords
    int randomX = random(0, width);
    int randomY = random(-30, -50);
    int randomW = random(8,14);
    int randomT = random(20,255);

    // store data
    neveX = append(neveX, randomX);
    neveY = append(neveY, randomY);
    neveW = append(neveW, randomW);
    neveT = append(neveT, randomT);
    neveTime = append(neveTime, random(0,50));
    neveDirection = append(neveDirection, -1);
  */

```



```
neveSpeed = append(neveSpeed, 1);
neveColor = append(neveColor, 283);
} */
}
```

```
////////////////////////////////////
// Draw
////////////////////////////////////
```

```
void draw() {

// LOGO SCREEN //////////////////////////////////////
//// the logo automatically fades after tot frames

if(sezione == START_SCREEN && mode == LOGO_SCREEN) {
  if(countseconds <= 90) ++countseconds;
  else countseconds = 0;
  if(countseconds == 90) {
    sezione = REGISTRATION;
    mode = LOGIN;
    myPhone.vibrate(800);          // spook vibrates!
  }
}
```

```
image(sfondo, 0, 0);          // background
```

```
// upper bar
textAlign(LEFT);
textFont(din_light_12_light_brown);
text("Spook", 6, 13);          // application title
textAlign(RIGHT);
text(totPunti+" pt", 234, 13); // points
if(totPunti < 0) totPunti = 0;
```

```
// MENU //////////////////////////////////////
```

```
if(viewMenu) {
  drawMenu(mode);
}
else {
  // STARTING SCREENS //////////////////////////////////////
  if(gameMode == 0) {
    drawScreens(sezione, mode);
  }
  // SINGLE PLAYER //////////////////////////////////////
  else if(gameMode == MODE_SINGLE_PLAYER) {
    drawSinglePlayer(sezione, mode);
  }
  // TIME CHALLENGE //////////////////////////////////////
  else if(gameMode == MODE_TIME_CHALLENGE) {
    drawTimeChallenge(sezione, mode);
  }
  // MULTIPLAYER //////////////////////////////////////
  else if(gameMode == MODE_MULTIPLAYER) {
    drawMultiplayer(sezione, mode);
  }
  // GALLERY //////////////////////////////////////
  else if(gameMode == MODE_GALLERY) {
    drawGallery(sezione, mode);
  }
}
}
```

```
////////////////////////////////////
// Keypad Event
```

```

////////////////////////////////////
void keyPressed() {
  if(keyCode == FIRE) {
    k=0;
    j=0;
    i=0;
  }

  //
  *****
  // ** MENU
  *****
  //
  *****
  // ATTIVA MENU *****
  if(activeMenu) {
    if(keyCode == SOFTKEY1) {
      reset_coriandoli();
      //reset_neve();
      viewMenu = true;
      modeMenu = BACK_TO_GAME;
    }
  }

  // VISUAL MENU *****
  if(viewMenu) {
    if(modeMenu == BACK_TO_GAME) {
      if(keyCode == DOWN) {
        modeMenu = GALLERY;
        return;
      }
      if(keyCode == FIRE) {
        viewMenu = false;
        return;
      }
    }
    if(modeMenu == GALLERY) {
      if(keyCode == DOWN) {
        modeMenu = RESTART_THE_GAME;
        return;
      }
      if(keyCode == UP) {
        modeMenu = BACK_TO_GAME;
        return;
      }
      if(keyCode == FIRE) {
        gameMode = MODE_GALLERY;
        sezione = MUSEUM_SELECTION;
        mode = 0;
        viewMenu = false;
        return;
      }
    }
    if(modeMenu == RESTART_THE_GAME) {
      if(keyCode == DOWN) {
        modeMenu = EXIT;
        return;
      }
      if(keyCode == UP) {
        modeMenu = GALLERY;
        return;
      }
    }
    if(keyCode == FIRE) {
      sezione = PLAY_OR_GALLERY;
      mode = PLAY_THE_GAME;
      gameMode = 0;
      viewMenu = false;
      return;
    }
  }
}

```

```

    }
}
if(modeMenu == EXIT) {
    if(keyCode == UP) {
        modeMenu = RESTART_THE_GAME;
        return;
    }
    if(keyCode == FIRE) {
        exit();
        return;
    }
}
}
else {

    //
    *****
    // ** INTRO
    *****
    //
    *****

    // REGISTRATION *****
    if(sezione == REGISTRATION) {

        // LOGIN //////////////////////////////////////
        if(mode == LOGIN) {
            nickname = multitapText; // sets nickname's value as text entered by the phone
            if(keyCode == FIRE) {
                nickname = nickname;
                mode = REGISTRATION_POPUP;
                return;
            }
        }

        // POP-UP //////////////////////////////////////
        if(mode == REGISTRATION_POPUP) {
            if(keyCode == SOFTKEY2) {
                mode = LOGIN;
                return;
            }
            if(nickname.length() > 0) { // if a nickname has been set, it gives u an alert
                if(keyCode == FIRE) {
                    sezione = STORY_01;
                    mode = HI_IM_SPOOK;
                    return;
                }
            }
        }
    }
}

// STORY_01 *****
if(sezione == STORY_01) {
    // HI IM SPOOK //////////////////////////////////////
    if(mode==HI_IM_SPOOK) {
        if(nickname.length() > 0) {
            if(keyCode == FIRE) {
                mode = IM_DETECTIVE;
                return;
            }
        }
    }
}
// IM A DETECTIVE //////////////////////////////////////
if(mode==IM_DETECTIVE) {
    if(nickname.length() > 0) {
        if(keyCode == FIRE) {
            mode = HELP_GHOST;
            return;
        }
    }
}

```

```

    }
}
// HELP GHOST //////////////////////////////////////
if(mode==HELP_GHOST) {
    if(nickname.length() > 0) {
        if(keyCode == FIRE) {
            mode = REST_IN_PEACE;
            return;
        }
    }
}
// REST IN PEACE //////////////////////////////////////
if(mode==REST_IN_PEACE) {
    if(nickname.length() > 0) {
        if(keyCode == FIRE) {
            mode = WOULD_YOU_HELP;
            return;
        }
    }
}
// WOULD U HELP //////////////////////////////////////
if(mode==WOULD_YOU_HELP) {
    if(nickname.length() > 0) {
        if(keyCode == FIRE) {
            sezione = PLAY_OR_GALLERY;
            mode = PLAY_THE_GAME;
            return;
        }
    }
}
}

// PLAY_OR_GALLERY *****
if(sezione == PLAY_OR_GALLERY) {
    if(mode==PLAY_THE_GAME) {
        if(keyCode == DOWN) {
            mode = VIEW_GALLERY;
            return;
        }
        if(keyCode == FIRE) {
            gameMode = 0;
            sezione = MUSEUM_LOCALIZATION;
            return;
        }
    }
}
if(mode==VIEW_GALLERY) {
    if(keyCode == UP) {
        mode = PLAY_THE_GAME;
        return;
    }
    if(keyCode == FIRE) {
        sezione = MUSEUM_SELECTION;
        gameMode = MODE_GALLERY;
        mode = 0;
        return;
    }
}
}

// MUSEUM LOCALIZATION *****
if(sezione == MUSEUM_LOCALIZATION) {
    if(keyCode == FIRE) {
        sezione = MENU_GAME_MODE;
        mode = MODE_SINGLE_PLAYER;
        return;
    }
}
}

// SELECT GAME MODE *****
if(sezione == MENU_GAME_MODE) {

```

```

// SINGLE PLAYER ///////////////////////////////////
if(mode == MODE_SINGLE_PLAYER) {
    if(keyCode == RIGHT) {
        mode = MODE_TIME_CHALLENGE;
        return;
    }
}

// TIME_CHALLENGE ///////////////////////////////////
if(mode == MODE_TIME_CHALLENGE) {
    if(keyCode == LEFT) {
        mode = MODE_SINGLE_PLAYER;
        return;
    }
    if(keyCode == RIGHT) {
        mode = MODE_MULTIPLAYER;
        return;
    }
}

// MULTIPLAYER ///////////////////////////////////
if(mode == MODE_MULTIPLAYER) {
    if(keyCode == LEFT) {
        mode = MODE_TIME_CHALLENGE;
        return;
    }
}

if(keyCode == FIRE) {
    gameMode = mode; // setto il gameMode secondo al schermata single/time/multi
selezionata
    sezione = NARRATIVE;
    mode = IN_THIS_MUSEUM;
    reset_paintings();
    println(paintings_found[0]+"-"+paintings_found[1]+"-"+paintings_found[2]+"-
"+paintings_found[3]); // resetto tutti i quadri e le risposte già date
    myPhone.vibrate(800);
    return;
}
}

```

```

//
*****
// ** SINGLE PLAYER
*****
//
*****
if(gameMode == MODE_SINGLE_PLAYER || gameMode == MODE_TIME_CHALLENGE) {

// NARRATIVE
*****
if(sezione == NARRATIVE) {

// IN_THIS_MUSEUM ///////////////////////////////////
if(mode == IN_THIS_MUSEUM) {
    if(keyCode == FIRE) {
        mode = MARCO_POLO_LEFT;
        return;
    }
}
// MARCO_POLO_LEFT ///////////////////////////////////
if(mode == MARCO_POLO_LEFT) {
    if(keyCode == FIRE) {
        mode = RECOGNIZE_CLUES;
        return;
    }
}
}

```

```

    }
}
// RECOGNIZE_CLUES //////////////////////////////////////
if(mode == RECOGNIZE_CLUES) {
    if(keyCode == FIRE) {
        mode = AT_THE_END;
        return;
    }
}
// AT_THE_END //////////////////////////////////////
if(mode == AT_THE_END) {
    if(keyCode == FIRE) {
        mode = DONT_GET_WRONG;
        return;
    }
}
// DONT_GET_WRONG //////////////////////////////////////
if(mode == DONT_GET_WRONG) {
    if(keyCode == FIRE) {
        if(gameMode == MODE_TIME_CHALLENGE){
            mode = TIME;
        }
        else if(gameMode == MODE_SINGLE_PLAYER){
            mode = READY;
        }
    }
    return;
}
}
// TIME //////////////////////////////////////
if(mode == TIME) {
    if(keyCode == FIRE) {
        mode = TIME_READY;
        return;
    }
}
// TIME_ENTER_A_ROOM //////////////////////////////////////
if(mode == TIME_ENTER_A_ROOM) {
    if(keyCode == FIRE) {
        mode = STAND_BY;
        return;
    }
}
// TIME_READY //////////////////////////////////////
if(mode == TIME_READY) {
    if(keyCode == FIRE) {
        mode = TIME_ENTER_A_ROOM;
        return;
    }
}
// READY //////////////////////////////////////
if(mode == READY) {
    if(keyCode == FIRE) {
        mode = ENTER_A_ROOM;
        return;
    }
}
// ENTER_A_ROOM //////////////////////////////////////
if(mode == ENTER_A_ROOM) {
    if(keyCode == FIRE) {
        mode = STAND_BY;
        return;
    }
}
}
// STAND_BY //////////////////////////////////////
if(mode == STAND_BY) {
    if(keyCode == FIRE) {
        sezione = MUSEUM_ROOM;
        mode = PAINT_1;
        if(gameMode == MODE_TIME_CHALLENGE){

```

```

        starTime = millis();
    }
    return;
}
}
}

// MUSEUM_ROOM
*****
if(sezione == MUSEUM_ROOM) {

    // 1 ////////////////////////////////////////////////////
    if(mode == PAINT_1) {

        if(paintings_found[PAINT_2] == 0) { // blocco la selezione
            if(keyCode == DOWN) {
                mode = PAINT_2;
                return;
            }
        }
        else if(paintings_found[PAINT_3] == 0) { // blocco la selezione
            if(keyCode == DOWN) {
                mode = PAINT_3;
                return;
            }
        }
        else if(paintings_found[PAINT_4] == 0) { // blocco la selezione
            if(keyCode == DOWN) {
                mode = PAINT_4;
                return;
            }
        }
    }
}

// 2 ////////////////////////////////////////////////////
if(mode == PAINT_2) {
    if(paintings_found[PAINT_3] == 0) { // blocco la selezione
        if(keyCode == DOWN) {
            mode = PAINT_3;
            return;
        }
    }
    else if(paintings_found[PAINT_4] == 0) { // blocco la selezione
        if(keyCode == DOWN) {
            mode = PAINT_4;
            return;
        }
    }
    if(paintings_found[PAINT_1] == 0) { // blocco la selezione
        if(keyCode == UP) {
            mode = PAINT_1;
            return;
        }
    }
}

// 3 ////////////////////////////////////////////////////
if(mode == PAINT_3) {
    if(paintings_found[PAINT_4] == 0) { // blocco la selezione
        if(keyCode == DOWN) {
            mode = PAINT_4;
            return;
        }
    }
    if(paintings_found[PAINT_2] == 0) { // blocco la selezione
        if(keyCode == UP) {
            mode = PAINT_2;
            return;
        }
    }
}
}
}

```

```

    }
    else if(paintings_found[PAINT_1] == 0) { // blocco la selezione
        if(keyCode == UP) {
            mode = PAINT_1;
            return;
        }
    }
}

// 4 //////////////////////////////////////
if(mode == PAINT_4) {
    if(paintings_found[PAINT_3] == 0) { // blocco la selezione
        if(keyCode == UP) {
            mode = PAINT_3;
            return;
        }
    }
    else if(paintings_found[PAINT_2] == 0) { // blocco la selezione
        if(keyCode == UP) {
            mode = PAINT_2;
            return;
        }
    }
    else if(paintings_found[PAINT_1] == 0) { // blocco la selezione
        if(keyCode == UP) {
            mode = PAINT_1;
            return;
        }
    }
}

// SELEZIONA QUADRO
if(keyCode == FIRE) {
    sezione = ICONS;
    modeIcon = SKULL;
    return;
}

// ICONS *****
if(sezione == ICONS) {
    if(keyCode == SOFTKEY2) {
        sezione = MUSEUM_ROOM;
        return;
    }

    if(keyCode == FIRE) {
        sezione = GOT_CLUES;
        return;
    }
}

// SKULL
if(modeIcon == SKULL) {
    if(icons_painting[mode][FOOD] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == RIGHT) {
            modeIcon = FOOD;
            return;
        }
    }
    else if(icons_painting[mode][SHIP] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == RIGHT) {
            modeIcon = SHIP;
            return;
        }
    }
}
if(icons_painting[mode][WEAPON] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == DOWN) {
        modeIcon = WEAPON;
        return;
    }
}

```



```

    }
  }
  else if(icons_painting[mode][FLAG] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == DOWN) {
      modeIcon = FLAG;
      return;
    }
  }
}

// FOOD
if(modeIcon == FOOD) {
  if(icons_painting[mode][SHIP] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == RIGHT) {
      modeIcon = SHIP;
      return;
    }
  }
  if(icons_painting[mode][SKULL] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == LEFT) {
      modeIcon = SKULL;
      return;
    }
  }
  if(icons_painting[mode][JEWEL] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == DOWN) {
      modeIcon = JEWEL;
      return;
    }
  }
  else if(icons_painting[mode][INSTRUMENT] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == DOWN) {
      modeIcon = INSTRUMENT;
      return;
    }
  }
}

// SHIP
if(modeIcon == SHIP) {
  if(icons_painting[mode][FOOD] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == LEFT) {
      modeIcon = FOOD;
      return;
    }
  }
  else if(icons_painting[mode][SKULL] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == LEFT) {
      modeIcon = SKULL;
      return;
    }
  }
  if(icons_painting[mode][PLANT] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == DOWN) {
      modeIcon = PLANT;
      return;
    }
  }
  else if(icons_painting[mode][ANIMAL] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == DOWN) {
      modeIcon = ANIMAL;
      return;
    }
  }
}

// WEAPON
if(modeIcon == WEAPON) {
  if(icons_painting[mode][JEWEL] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == RIGHT) {

```

```

        modeIcon = JEWEL;
        return;
    }
}
else if(icons_painting[mode][PLANT] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == RIGHT) {
        modeIcon = PLANT;
        return;
    }
}
if(icons_painting[mode][SKULL] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == UP) {
        modeIcon = SKULL;
        return;
    }
}
if(icons_painting[mode][FLAG] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == DOWN) {
        modeIcon = FLAG;
        return;
    }
}
}

// JEWEL
if(modeIcon == JEWEL) {
    if(icons_painting[mode][PLANT] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == RIGHT) {
            modeIcon = PLANT;
            return;
        }
    }
    if(icons_painting[mode][WEAPON] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == LEFT) {
            modeIcon = WEAPON;
            return;
        }
    }
    if(icons_painting[mode][FOOD] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == UP) {
            modeIcon = FOOD;
            return;
        }
    }
    if(icons_painting[mode][INSTRUMENT] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == DOWN) {
            modeIcon = INSTRUMENT;
            return;
        }
    }
}

// PLANT
if(modeIcon == PLANT) {
    if(icons_painting[mode][JEWEL] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == LEFT) {
            modeIcon = JEWEL;
            return;
        }
    }
}
else if(icons_painting[mode][WEAPON] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == LEFT) {
        modeIcon = WEAPON;
        return;
    }
}
if(icons_painting[mode][SHIP] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == UP) {
        modeIcon = SHIP;
        return;
    }
}

```

```

    }
}
if(icons_painting[mode][ANIMAL] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == DOWN) {
        modeIcon = ANIMAL;
        return;
    }
}
}

// FLAG
if(modeIcon == FLAG) {
    if(icons_painting[mode][INSTRUMENT] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == RIGHT) {
            modeIcon = INSTRUMENT;
            return;
        }
    }
    else if(icons_painting[mode][ANIMAL] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == RIGHT) {
            modeIcon = ANIMAL;
            return;
        }
    }
}
if(icons_painting[mode][WEAPON] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == UP) {
        modeIcon = WEAPON;
        return;
    }
}
else if(icons_painting[mode][SKULL] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == UP) {
        modeIcon = SKULL;
        return;
    }
}
}

// INSTRUMENT
if(modeIcon == INSTRUMENT) {
    if(icons_painting[mode][ANIMAL] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == RIGHT) {
            modeIcon = ANIMAL;
            return;
        }
    }
}
if(icons_painting[mode][JEWEL] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == UP) {
        modeIcon = JEWEL;
        return;
    }
}
else if(icons_painting[mode][FOOD] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == UP) {
        modeIcon = FOOD;
        return;
    }
}
}
if(icons_painting[mode][FLAG] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == LEFT) {
        modeIcon = FLAG;
        return;
    }
}
}

// ANIMAL
if(modeIcon == ANIMAL) {
    if(icons_painting[mode][PLANT] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == UP) {

```

```

        modeIcon = PLANT;
        return;
    }
}
else if(icons_painting[mode][SHIP] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == UP) {
        modeIcon = SHIP;
        return;
    }
}
if(icons_painting[mode][INSTRUMENT] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == LEFT) {
        modeIcon = INSTRUMENT;
        return;
    }
}
else if(icons_painting[mode][FLAG] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == LEFT) {
        modeIcon = FLAG;
        return;
    }
}
}
}

// TAKE PHOTO
*****
if(sezione == TAKE_PHOTO) {
    if(keyCode == SOFTKEY2) {
        hideCamera();
        sezione = ICONS;
        return;
    }
    if(keyCode == FIRE) {
        //readCamera();
        hideCamera();
        sezione = GOT_CLUES;
        return;
    }
}

// GOT CLUES
*****
if(sezione == GOT_CLUES) {
    if(keyCode == FIRE) {
        //reset_neve();

        // setto l'icona interessata come già trovata (quindi non selezionabile)
        if(icons_painting[mode][modeIcon] == 1) {
            totPunti = totPunti + 10;
            icons_painting[mode][modeIcon] = 2;
        }
        else { // controllo quanti indizi aveva trovato e tolgo i punti!!!
            for(i=0;i<=8;i++) {
                if(icons_painting[mode][i] == 2) {
                    totPunti = totPunti - 10;
                    if(totPunti < 0) totPunti = 0;
                }
            }
        }
    }

    // se ci sono ancora indizi da trovare E se non ho sbagliato un indizio, torno al
quadro...
    if(clues > 1 && paintings_found[mode] == 0) {
        sezione = ICONS;
        // controllo quali icone sono ancora 'disponibili' nella lista, e metto il focus su
una di esse
        for(i=0;i<=8;i++) {
            if(icons_painting[mode][i] < 2) {
                modeIcon = i;
            }
        }
    }
}

```

```

        break;
    }
}
}
else { //... se no torno alla stanza dei musei!
    if(clues==1) {
        paintings_found[mode] = 1;    // se ho trovato tutti gli indizi, disattivo il quadro
selezionato
    }

    // controllo quali quadri sono ancora 'disponibili' nella lista, e metto il focus su
uno di essi
    if(paintings_found[PAINT_1] == 0) {
        mode = PAINT_1;
        sezione = MUSEUM_ROOM;
    }
    else if(paintings_found[PAINT_2] == 0) {
        mode = PAINT_2;
        sezione = MUSEUM_ROOM;
    }
    else if(paintings_found[PAINT_3] == 0) {
        mode = PAINT_3;
        sezione = MUSEUM_ROOM;
    }
    else if(paintings_found[PAINT_4] == 0) {
        mode = PAINT_4;
        sezione = MUSEUM_ROOM;
    }
    // se son passati tutti, vado alla schermata di STANZA COMPLETATA
    else {
        sezione = ROOM_COMPLETE;
        myPhone.vibrate(800);        // spook vibrates!
        return;
    }
}
return;
}
}

// ROOM_COMPLETE
*****
if(sezione == ROOM_COMPLETE) {
    if(keyCode == FIRE) {
        reset_coriandoli();
        sezione = STAND_BY_TO_EXIT;
        return;
    }
}

// STAND BY TO EXIT
*****
if(sezione == STAND_BY_TO_EXIT) {
    if(keyCode == FIRE) {
        sezione = EXIT_MUSEUM;
        myPhone.vibrate(800);        // spook vibrates!
        return;
    }
}

// EXIT MUSEUM
*****
if(sezione == EXIT_MUSEUM) {
    if(keyCode == FIRE) {
        sezione = FINAL_QUESTION;
        mode = QUESTION_1;
        modeIcon = SKULL;
        return;
    }
}
}

```

```

// FINAL QUESTION
*****
if(sezione == FINAL_QUESTION) {
println("modeXbarra="+modeXbarra);
println("scrollingText="+scrollingText);
println("mode="+mode);
// con up/down muovo il testo
if(keyCode == UP) {
    if(scrollingText > mode*-1 && scrollingText <= 0) {
        scrollingText = scrollingText -1;
        modeXbarra = modeXbarra -1;
    }
}
if(keyCode == DOWN) {
    if(scrollingText >= mode*-1 && scrollingText < 0) {
        scrollingText = scrollingText +1;
        modeXbarra = modeXbarra +1;
    }
}
// con destra e sinistra mi muovo nella barra delle icone
if(keyCode == LEFT) {
    if(modeIcon>0) modeIcon = modeIcon-1;
    return;
}
if(keyCode == RIGHT) {
    if(modeIcon<8) modeIcon = modeIcon+1;
    return;
}
if(keyCode == FIRE) {
    if(mode < QUESTION_6) {
        // setto la risposta
        given_answers[mode] = modeIcon;
        // punti
        if(mode > QUESTION_1) {
            if(given_answers[mode] == correct_answers[mode]) totPunti += 30;
            else totPunti -= 20;
        }

        mode += 1;          // passo alla prox question (a meno che nn sia già all'ultima)
        modeXbarra = mode;
        scrollingText = 0;
        return;
    }
    if(mode == QUESTION_6) {
        sezione = LAST_SCREEN;
        mode = CASE_SOLVED;
        //myPhone.vibrate(800);          // spook vibrates!
        myPhone.vibrate(100);          // final vibration!!!
        myPhone.vibrate(100);
        myPhone.vibrate(100);
        myPhone.vibrate(800);
        return;
    }
}
}
// LAST_SCREEN
*****
if(sezione == LAST_SCREEN) {
    if(keyCode == FIRE){
        if(mode==CASE_SOLVED) {
            mode = GAME_COMPLETE;
            return;
        }
    }
}
}
}

```

```

//
*****
// ** GALLERY *****
//
*****
if(gameMode == MODE_GALLERY) {

    // GALLERY: SELECT MUSEUM *****
    if(sezione == MUSEUM_SELECTION) {

        // ACCADEMIA //////////////////////////////////
        if(mode == ACCADEMIA) {
            if(keyCode == RIGHT) {
                mode = CORRER;
                return;
            }
        }

        // CORRER //////////////////////////////////
        if(mode == CORRER) {
            if(keyCode == LEFT) {
                mode = ACCADEMIA;
                return;
            }
            if(keyCode == RIGHT) {
                mode = CAPESARO;
                return;
            }
        }

        // CA PESARO //////////////////////////////////
        if(mode == CAPESARO) {
            if(keyCode == LEFT) {
                mode = CORRER;
                return;
            }
        }

        // SELEZIONA MUSEO
        if(keyCode == FIRE) {
            sezione = ICONS;
            modeIcon = SKULL;
            return;
        }
    }
}

// GALLERY ICONS
*****
if(sezione == ICONS) {
    if(keyCode == FIRE) {
        sezione = PICS_BROWSING;
        return;
    }

    // SKULL
    if(modeIcon == SKULL) {
        if(icons_painting[mode][FOOD] < 2) { // blocco l'icona se è già stata usata
            if(keyCode == RIGHT) {
                modeIcon = FOOD;
                return;
            }
        }
    }
    else if(icons_painting[mode][SHIP] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == RIGHT) {
            modeIcon = SHIP;
            return;
        }
    }
}

```

```

}
if(icons_painting[mode][WEAPON] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == DOWN) {
        modeIcon = WEAPON;
        return;
    }
}
else if(icons_painting[mode][FLAG] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == DOWN) {
        modeIcon = FLAG;
        return;
    }
}
}

// FOOD
if(modeIcon == FOOD) {
    if(icons_painting[mode][SHIP] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == RIGHT) {
            modeIcon = SHIP;
            return;
        }
    }
    if(icons_painting[mode][SKULL] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == LEFT) {
            modeIcon = SKULL;
            return;
        }
    }
    if(icons_painting[mode][JEWEL] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == DOWN) {
            modeIcon = JEWEL;
            return;
        }
    }
    else if(icons_painting[mode][INSTRUMENT] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == DOWN) {
            modeIcon = INSTRUMENT;
            return;
        }
    }
}

// SHIP
if(modeIcon == SHIP) {
    if(icons_painting[mode][FOOD] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == LEFT) {
            modeIcon = FOOD;
            return;
        }
    }
    else if(icons_painting[mode][SKULL] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == LEFT) {
            modeIcon = SKULL;
            return;
        }
    }
    if(icons_painting[mode][PLANT] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == DOWN) {
            modeIcon = PLANT;
            return;
        }
    }
    else if(icons_painting[mode][ANIMAL] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == DOWN) {
            modeIcon = ANIMAL;
            return;
        }
    }
}
}

```



```

// WEAPON
if(modeIcon == WEAPON) {
    if(icons_painting[mode][JEWEL] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == RIGHT) {
            modeIcon = JEWEL;
            return;
        }
    }
    else if(icons_painting[mode][PLANT] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == RIGHT) {
            modeIcon = PLANT;
            return;
        }
    }
    if(icons_painting[mode][SKULL] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == UP) {
            modeIcon = SKULL;
            return;
        }
    }
    if(icons_painting[mode][FLAG] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == DOWN) {
            modeIcon = FLAG;
            return;
        }
    }
}

// JEWEL
if(modeIcon == JEWEL) {
    if(icons_painting[mode][PLANT] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == RIGHT) {
            modeIcon = PLANT;
            return;
        }
    }
    if(icons_painting[mode][WEAPON] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == LEFT) {
            modeIcon = WEAPON;
            return;
        }
    }
    if(icons_painting[mode][FOOD] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == UP) {
            modeIcon = FOOD;
            return;
        }
    }
    if(icons_painting[mode][INSTRUMENT] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == DOWN) {
            modeIcon = INSTRUMENT;
            return;
        }
    }
}

// PLANT
if(modeIcon == PLANT) {
    if(icons_painting[mode][JEWEL] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == LEFT) {
            modeIcon = JEWEL;
            return;
        }
    }
    else if(icons_painting[mode][WEAPON] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == LEFT) {
            modeIcon = WEAPON;
            return;
        }
    }
}

```

```

}
if(icons_painting[mode][SHIP] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == UP) {
        modeIcon = SHIP;
        return;
    }
}
if(icons_painting[mode][ANIMAL] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == DOWN) {
        modeIcon = ANIMAL;
        return;
    }
}
}

// FLAG
if(modeIcon == FLAG) {
    if(icons_painting[mode][INSTRUMENT] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == RIGHT) {
            modeIcon = INSTRUMENT;
            return;
        }
    }
    else if(icons_painting[mode][ANIMAL] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == RIGHT) {
            modeIcon = ANIMAL;
            return;
        }
    }
    if(icons_painting[mode][WEAPON] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == UP) {
            modeIcon = WEAPON;
            return;
        }
    }
    else if(icons_painting[mode][SKULL] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == UP) {
            modeIcon = SKULL;
            return;
        }
    }
}

// INSTRUMENT
if(modeIcon == INSTRUMENT) {
    if(icons_painting[mode][ANIMAL] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == RIGHT) {
            modeIcon = ANIMAL;
            return;
        }
    }
    if(icons_painting[mode][JEWEL] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == UP) {
            modeIcon = JEWEL;
            return;
        }
    }
    else if(icons_painting[mode][FOOD] < 2) { // blocco l'icona se è già stata usata
        if(keyCode == UP) {
            modeIcon = FOOD;
            return;
        }
    }
}
if(icons_painting[mode][FLAG] < 2) { // blocco l'icona se è già stata usata
    if(keyCode == LEFT) {
        modeIcon = FLAG;
        return;
    }
}
}
}

```



```
        if(keyCode == LEFT) {
            mode = FLAG_PIC;
            return;
        }
    }
}

// shortcut
if(keyCode=='#') {
    nickname = "Tommaso";
    gameMode = MODE_SINGLE_PLAYER;
    sezione = FINAL_QUESTION;
    mode = QUESTION_1;
    totPunti = 420;
    return;
}
if(keyCode=='0') {
    nickname = "Tommaso";
    sezione = PLAY_OR_GALLERY;
    mode = PLAY_THE_GAME;
    gameMode = 0;
}
}
```