بسم لله الرحمن الرحیم

پکیج CrossWalk :

توضیحات :

پکیج اصلی شامل کلاس اصلی و متد main

کلاس ها :

StartGame.java :‌ کلاس اجرا کننده ی برنامه که شامل متد main است .

-ارث بری ها :

- بدون ارث بری !

-فیلدها :

- ندارد !

-متدها :

- متد استاتیک main برای شروع برنامه .

پکیج CrossWalk.UI :

توضیحات :

پکیج جهت نمایش خروجی به کاربر و رابط کاربری

کلاس ها :

GameListener.java‌ : کلاس نگهدارنده ی بازخورد ها به رویداد های ورودی مثلا کلیک موس و وارد کردن کلید کیبورد

-ارث بری ها :

- بدون ارث بری !

-فیلدها :

- private final KeyListener KeyListener : نگهداری پاسخ به بازخورد های صفحه کلید

- private final MouseListener MouseListener : نگهداری پاسخ به بازخورد های موس

-متدها :

public GameListener(ArrayList<Line> Lines)- : سازنده ی کلاس با ورودی یک مراجعه به لاین های بازی جهت ذخیره در فایل و بازیابی در هنگام ادامه ی بازی

-public KeyListener getKeyListener() : متد کوئری با خروجی KeyListener

-public KeyListener getMouseListener() : متد کوئری با خروجی MouseListener

InitGame.java : کلاس تنظیم کننده ی نوع بازی مثلا بازی جدید ، مشاهده ی فیلم ،‌ حرکت خودکار ادمک ، بازی ذخیره شده

-ارث بری ها : ندارد

-فیلدها :

public static boolean GameStop- : فیلد استاتیک booleanبرای بررسی وضعیت توقف بازی

-public static boolean GameEnd : فیلد استاتیک booleanبرای بررسی وضعیت اتمام بازی

-متدها :

- InitGame() : سازنده ی پیشفرض بازی.

-autoMoveSheep() : پیاده سازی بازی براساس حرکت خودکار ادمک

-newGame() : پیاده سازی بازی جدید

-loadResumeGame() : پیاده سازی بازی براساس بازی قبلی و ادامه ی ان

- replyTheMovie() :‌ پیاده سازی فیلم بازی انجام شده.

InitGraphic.java : کلاس تنظیم کننده ی گرافیک بازی و بخش UI

-ارث بری ها :

- ارث بری از کلاس JPanel برای پیاده سازی گرافیک با استفاده از swing

-پیاده سازی اینترفیس Runnable برای پیاده سازی thread

-فیلدها :

private ArrayList<Line> Lines- : نگهدارنده ی ارجاع به لاین های بازی

-private final int MiddleOfCrosswalkPosition‌ :‌ نگهدارنده ی مختصات وسط خط کشی عابر پیاده

-private BufferedImage CrosswalkImage : نگهدانده ی عکس خط کشی عابرپیاده

-private BufferedImage LineImage : نگهدارنده ی عکس لاین ها

-private BufferedImage MiddleLineImage :‌نگهدارنده ی عکس خط بین لاین ها

-private final int TopLineCount : تعداد لاین های بالایی

-public static Sheep Sheep = new Sheep() :‌ فیلد استاتیک برای نگهداری شی از ادمک

-private BufferedImage HeartImage :‌ نگهدارنده ی عکس قلب برای نمایش life ادمک

-متدها :

- public InitGraphic(ArrayList<Line> Lines) : سازنده با ورودی مراجعه به لاین ها

-private void setInit() :‌ تنظیم کردن پنل نمایشی بازی مانند title,size و ...

-public void paintComponent(Graphics g) : متد override شده از کلاس پدر جهت نمایش شی ها

-public void run() : پیاده سازی runnable جهت ایجاد یک ترد برای حرکت دراوردن اشیا

پکیج CrossWalk.Menu :

توضیحات :

پکیج منو های بازی

کلاس ها :

Menu.java : کلاس abstract جهت ارث بری منوها و پیاده سازی فیلد ها و متدهای مشترک

* Show() :‌ متد نمایش منو
* setInit(JFrame frame) : انجام تنظیمات اولیه

MainMenu.java :‌ کلاس منوی اصلی

CarMenu.java : کلاس انتخاب ماشین در تنظیمات

LineMenu.java :‌ کلاس انتخاب لاین ها ، خط عابر پیاده ، خط بین لاین ها

ReplyMenu.java :‌ کلاس منوی نمایش دهنده ی بازی های انجام شده .

ResumeGameMenu.java :‌ کلاس منوی توقف در حین بازی

SelectMapMenu.java : کلاس منوی انتخاب مپ قبل از شروع بازی

SettingMenu.java :‌ کلاس جهت نمایش تنظیمات اصلی بازی

SheepMenu.java : کلاس برای انتخاب شخصیت بازی

GameSetting.java :‌ کلاس جهت ذخیره سازی تنظیمات.

SettingConst.java :‌ کلاس جهت ذخیره ی ثابت های بازی

MenuConst.java : کلاس نگهدارنده ی ثابت های منو

پکیج CrossWalk.StoreData :

توضیحات :

پکیج ذخیره سازی اطلاعات روی فایل

کلاس ها :

ResumeAndLoad.java :‌ کلاس ذخیره ساز داده های برای بخش ادامه ی بازی قبلی

-ارث بری: ندارد

فیلد : ندارد

متد ها :

-public ResumeAndLoad() :‌ سازنده ی پیشفرض

- public final void initForSaveGame(): تنظیمات اولیه برای ذخیره سازی مثلا ساخت فولدر Resume درصورت عدم وجود

- public boolean isExistResumeFile() :‌ بررسی اینکه ایا پوشه ی Resume موجود است؟

- public ArrayList<Line> loadGame() : برگرداندن لاین های ساخته شده از فایل موجود و مقدار دهی به فیلد های ادمک

- public void saveGame(ArrayList<Line> lines) :‌ ذخیره کردن لاین ها که از ورودی میگیرید در فایل

- private void writeLinesDataToFile(ArrayList<Line> lines) : نوشتن اطلاعات لاین ها در فایل

- private void writeSheepDataToFile() :‌ نوشتن اطلاعات ادمک در فایل ها

-public void SaveGameForResume(ArrayList<Line> line) : سرویس دهنده به کلاس استفاده کننده و استفاده از دو متد بالا

WriteReplyData.java :‌ کلاس ذخیره ساز داده های مورد نیاز جهت نمایش فیلم بازی انجام شده

-ارث بری: ندارد

-فیلد ها‌:

-private LocalDateTime StartedTime : نگهدارنده ی زمان جاری برای تنظیمات مربوط به حرکت خودکار ادک

-‌ private String Path : نگهدارنده ی ادرس فولدری که قرار است داده ها درون ان ریخته شوند.

-متد ها :

- public final void init() : ساختن ادرس با استفاده از زمان جهت یکتا شدن ان و ساخت فایل تنظیمات جهت نگهداری تنظیمات بازی

- public void appendCarsToFile(Car tempCar) : اضافه کردن اطلاعات ماشین ها به فایل

public void appendSheepToFile(int keyCode)- : اضافه کردن اطلاعات حرکت ادمک به فایل

پکیج CrossWalk.Utilities :

توضیحات :

پکیج شامل کلاس های مفید و استفاده شده در بازی

کلاس ها :

ExceptionWriter.java :‌ کلاس ذخیره خطاهای رخ داده در برنامه و لاگ گرفتن از انها

-ارث بری : ندارد

-فیلد‌: ندارد

-متده ها:

- public void write(String cause, Exception ex, boolean isImportante) : متدی برای لاگ گرفتن از خطا ها و نوشتن خطاها در فایل

Const.java : کلاس نگهدارنده ی تنظیمات ثابت بازی

- ارث بری : ندارد

- فیلد‌: شامل تعداد زیادی از ثابت های بازی که درطول برنامه همیشه ثابت هستند مثلا مقدار کم شدن سرعت هنگام رسیدن به خط عابر پیاده ویا ادرس پوشه ی عکسهای بازی

- متد ها :‌ ندارد

پکیج CrossWalk.AutoWork:

توضیحات :

پکیج شامل کلاس هایی که به صورت خودکار اقدام به انجام کار ها میکنندو اغلب یک thread پیاده سازی میکنند

کلاس ها :

CreateCar.java :‌ کلاس abstract جهت ساخت ماشین های بازی و پیاده سازی اطلاعات لاین ها

-ارث بری :

-اینترفیس Runnable جهت پیاده سازی Thread

-فیلدها:

- private ArrayList<Line> Lines: نگهدارنده ی یک مراجعه به لاین های بازی

-private final int RtlLineCount :‌ نگهدارنده ی تعداد لاین های راست به چپ

-private final int LtrLineCount : نگهدارنده ی تعداد لاین های چپ به راست

-متد ها:

-public CreateCar() :‌ سازنده ی پیشفرض برای مقدار دهی اولیه ی تمام فیلدها

-public void InitLine(): متد سازنده ی لاین های بازی از فایل تنظیمات

-public void InitLineWithCars(ArrayList<Line> line) : متد سازنده ی لاین ها و اضافه کننده ی ماشین ها به لاین ها درزمان ادامه ی بازی

-public ArrayList<Line> getLine() : متد کوئری با خروجی لاین ها

-public int getRtlLineCount() : متد کوئری با خروجی تعداد لاین های راست به چپ

-public int getLtrLineCount() : متد کوئری با خروجی تعداد لاین های چپ به راست.

- public final void setLine(ArrayList<Line> line) : متد set کننده ی فیلد لاین

- public abstract void run(): پیاده سازی متد اینترفیس جهت ایجاد Thead و سپردن بدنه ی تابع به فرزندان با کلمه ی کلیدی abstract

CreateCarInNewGame.java : ساخت ماشین ها به صورت رندوم در بازی جدید

-ارث بری :

-ارث بری از کلاس abstract با نام CreateCar

-پیاده سازی اینترفیس Runnable که پیاده سازی ان از پدر به فزرند سپرده شده است

-فیلد ها :

- public boolean SaveReply : نگهدارنده ی این که ایا داده ها برای فیلم ذخیره شوند؟

- public WriteReplyData WriteReplyData :‌ نگهداری کلاسی که جهت نوشتن اطلاعات برای فیلم بازی نیاز است در واقع پیاده سازی رابطه ی Has a

-متدها :

- public CreateCarInNewGame(boolean SaveReply) : سازنده با ورودی اینکه ایا داده ها برای فیلم ذخیره شوند یا خیر . درصورت true بودن مقدار دهی فیلد دیگر

- public void run() : override متد پدر برای ایجاد Thread

CreateCarInReply.java : ساخت ماشین ها با اطلاعات قبلی در زمان نمایش فیلم

-ارث بری :

-ارث بری از کلاس abstract با نام CreateCar

-پیاده سازی اینترفیس Runnable که پیاده سازی ان از پدر به فزرند سپرده شده است

-فیلدها‌:

- private final String CarsPath نگهدارنده ی ادرس اطلاعات ذخیره شده ی ماشین ها

-متدها:

-public CreateCarInReply(String CarsPath) :‌ سازنده ی کلاس و مقدار دهی اولیه ی ادرس ماشین ها

- public void run() : override متد پدر برای ایجاد Thread

SheepMoverInBotGame.java :‌ کلاس حرکت دهنده ی ادمک در بازی خودکار ادمک

-ارث بری :

-پیاده سازی اینترفیس Runnable برای ایجاد Thread

-فیلد ها :

-private final long SleepInMilliSecond : زمان بین دو حرکت ادمک

-private final int RandRate: نرخ حرکت به جلو

-متدها :

-public SheepMoverInBotGame(long SleepInMilliSecond, int RandRate) : سازنده ی کلاس و مقدار دهی به فیلدها با ورودی

-public void run() : پیاده سازی متد اینترفیس با Override کردن ان و ایجاد یک Thread

SheepMoverInReply.java : کلاس حرمت دهنده ی خودکار ادمک در مشاهده ی فیلم

-ارث بری ها :

- اینترفیس Runnable جهت پیاده سازی Thread

-فیلد ها :

-private final String SheepPath : ادرس فایل نگهدارنده ی اطلاعات حرکت ادمک

-متدها :

-public SheepMoverInReply(String SheepPath) : سازنده ی کلاس و مقدار دهی فیلد از ورودی

-public void run() : پیاده سازی متد اینترفیس با Override کردن ان و ایجاد یک Thread

پکیج CrossWalk.Object:

توضیحات :

پکیج شامل کلاس های تعریف کننده ی اشیا بازی

کلاس ها :

Drawable.java : اینترفیس جهت پیاده سازی قابلیت کشیده شدن و نمایش برای کاربر

-متد ها :‌

- public BufferedImage getImage(): متد کوئری با خروجی عکس شی

- public int getYPositionForDraw(): متد کوئری با خروجی مختصات yشی

-public int getXPositionForDraw()‌: متد کوئری با خروجی مختصات x شی

Moveable.java : اینترفیس جهت پیاده سازی قابلیت حرکت

-متدها :

- public void move() : متد حرکت دادن اشیا و تغییر مختصات انها

Line.java : کلاس لاین های بازی

-ارث بری :

- پیاده سازی انترفیس Serializable جهت نوشتن درون فایل

- فیلدها :

- private int Id : ایدی هر لاین

- private int MaxCarSpeed حداکثر سرعت ماشین ها درون لاین

- private int MinCarSpeed حداقل سرعت ماشین ها درون لاین

- private boolean Direction :‌ جهت لاین (ثابت مورد استفاده درکلاس Const موجود است)

- private int YPosition :‌ مختصات Y هر لاین برای نمایش (X ان سراسر پنجره است)

- private ArrayList<Car> Cars : ایجاد رابطه ی Has a و دسترسی به ماشین های هر لاین

- private float[] CrosswalkXPosition : مختصات شروع و پایان X برای خط عابر پیاده

-private int CreatedCarCount تعداد ماشین های ساخته شده برای هر لاین

- public static int SheepCurrentLine : لاینی که ادمک روی ان قرار گرفته است

- private boolean CanCarOvertaking : ایا در لاین امکان سبقت وجود دارد ؟ بررسی وجود لاین کناری مناسب

-متد ها :

-public Line() : سازنده ی پیشفرض

public Line(int Id, int MaxCarSpeed, int MinCarSpeed, boolean Direction, int YPosition, boolean CanCarOvertaking) : سازنده ی کلاس

public Line(int Id, int MaxCarSpeed, int MinCarSpeed, boolean Direction, int Position, boolean CanCarOvertaking, int CreatedCarCount) :‌سازنده ی کلاس

-public int getYPosition()‌ : برگرداندن مختصات Y برای لاین

-public boolean getCanCarOvertaking() :‌ برگرداندن اینکه ایا امکان سبقت هست ؟

-public int getCreatedCarCount() : برگرداندن تعداد ماشین های ساخته شده

-public int getMinCarSpeed() برگرداندن حداقل سرعت ماشین های لاین

-public int getMaxCarSpeed() برگرداندن حداکثر سرعت ماشین ها

-public ArrayList<Car> getCars() :‌برگرداندن ماشین های درون لاین

-public float[] getCrosswalkXPosition() : برگرداندن محدوده ی مختصات x خط کشی عابر پیاده

-public boolean getDirection() برگرداندن جهت لاین

-public int getId() :‌برگرداندن ایدی لاین

-public String toString() :‌ override کردن برای نمایش و ذخیره ی اطلاعات لاین ها

-public void addCar(Car newCar) : متد اضافه کردن ماشینی که از قبل ساخته شده

-public boolean createNewCar(Car newCar) :‌متد اضافه کردن ماشین جدید به لاین

-public void disposeCar(Car car) متد پاک کردن ماشین از لاین

-private float[] calcuteXCrosswalkPosistion() ‌: محاسبه ی محدوده ی x خط کشی عابرپیاده

-public void increaseCreatedCarCount() اضافه کردن یک واحد به تعداد ماشین های ساخته شده ی لاین

Car.java : کلاس abstract ماشین های بازی

CarRtl.java : کلاس ماشین هایی که جهت انها از راست به چپ است

CarLtr.java‌ : کلاس ماشین های که جهت انها از چپ به راست است

CarType.java : کلاسی که نوع ماشین را بیان میکند

Sheep.java :‌ کلاس ادمک بازی