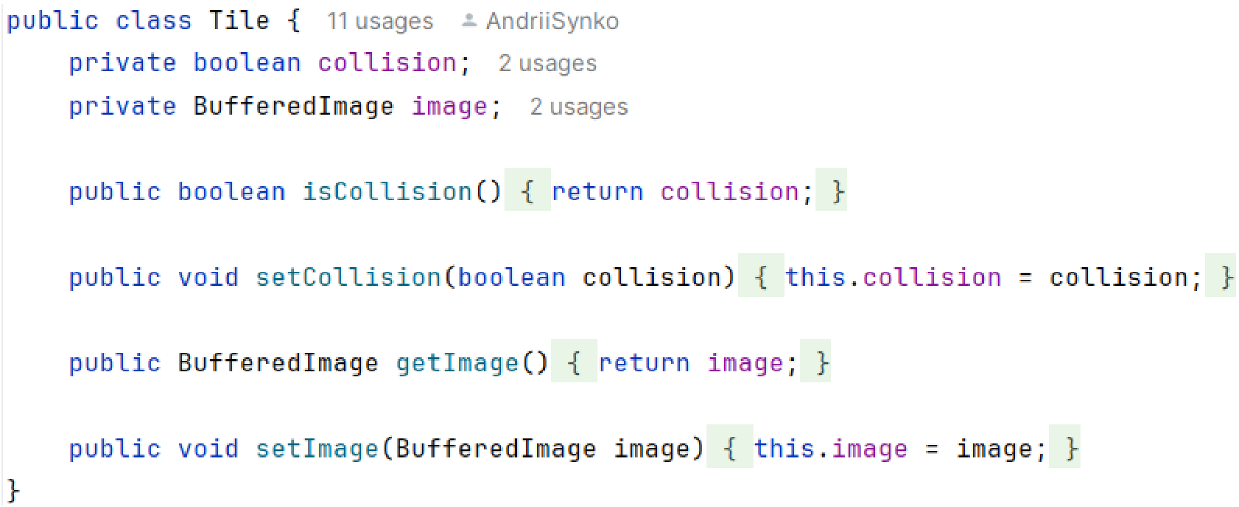
Andrii Synko

OOP Principles

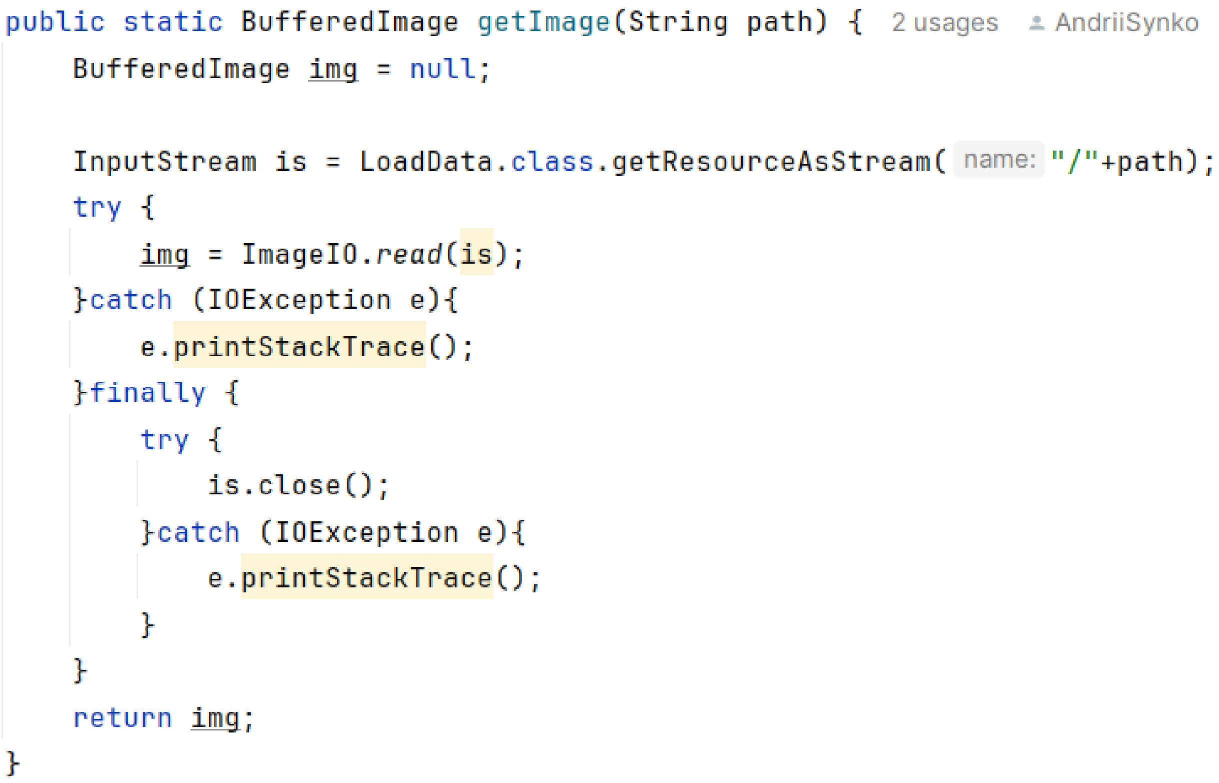
Encapsulation:

- Via private attributes and mutators
  Where: src/map/Tile.java:5-24
  Code:



```
public class Tile {  11 usages  AndriiSynko
    private boolean collision;  2 usages
    private BufferedImage image;  2 usages

    public boolean isCollision() { return collision; }

    public void setCollision(boolean collision) { this.collision = collision; }

    public BufferedImage getImage() { return image; }

    public void setImage(BufferedImage image) { this.image = image; }
}
```

- Via static methods? (does not violate encapsulation, but does not hiding method itself)
  Where: src/util/LoadData.java:15-31

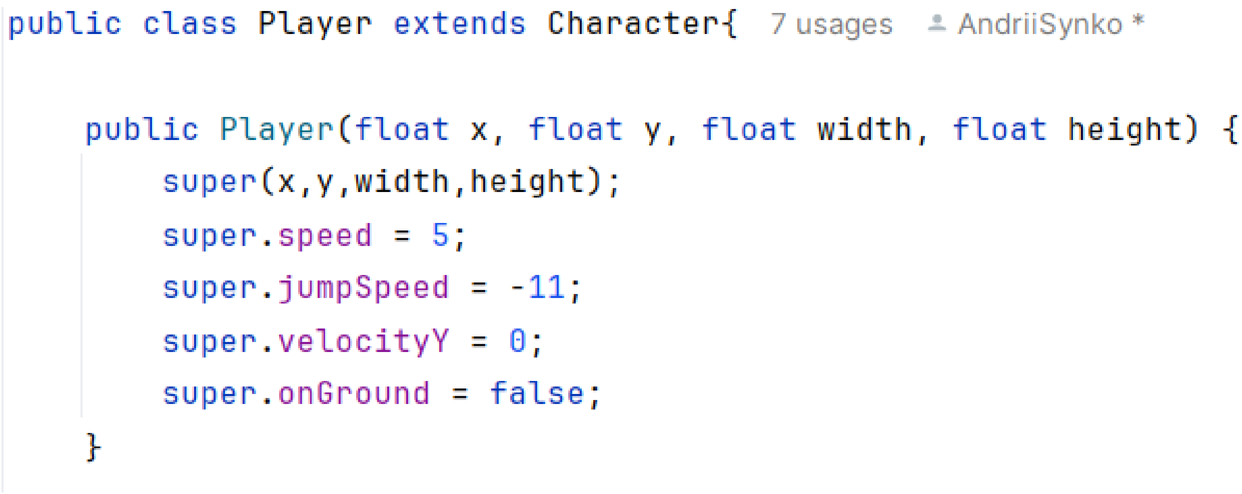


```
public static BufferedImage getImage(String path) {  2 usages  AndriiSynko
    BufferedImage img = null;

    InputStream is = LoadData.class.getResourceAsStream( name: "/"+path);
    try {
        img = ImageIO.read(is);
    }catch (IOException e){
        e.printStackTrace();
    }finally {
        try {
            is.close();
        }catch (IOException e){
            e.printStackTrace();
        }
    }
    return img;
}
```

Andrii Synko

Inheritance:

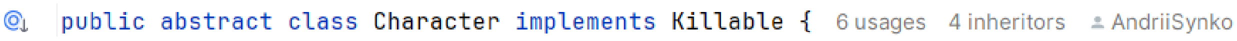
Where: src/entity/Player.java:13-23



```
public class Player extends Character{  7 usages  AndriiSynko *

    public Player(float x, float y, float width, float height) {
        super(x,y,width,height);
        super.speed = 5;
        super.jumpSpeed = -11;
        super.velocityY = 0;
        super.onGround = false;
    }
```

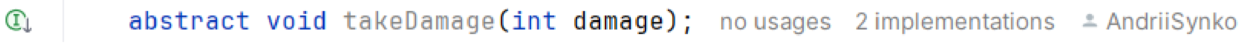
Abstraction:

Where: src/entity/Character.java:6



```
public abstract class Character implements Killable {  6 usages  4 inheritors  AndriiSynko
```

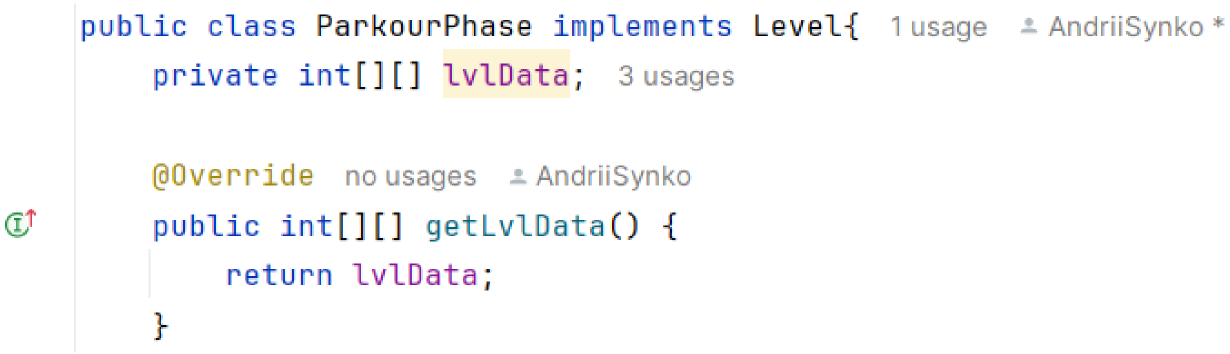
- Abstract methods were defined, but not implemented in this ckeckpoint
  Where: src/entity/Character.java:39



```
    abstract void takeDamage(int damage);  no usages  2 implementations  AndriiSynko
```

Polymorphism:

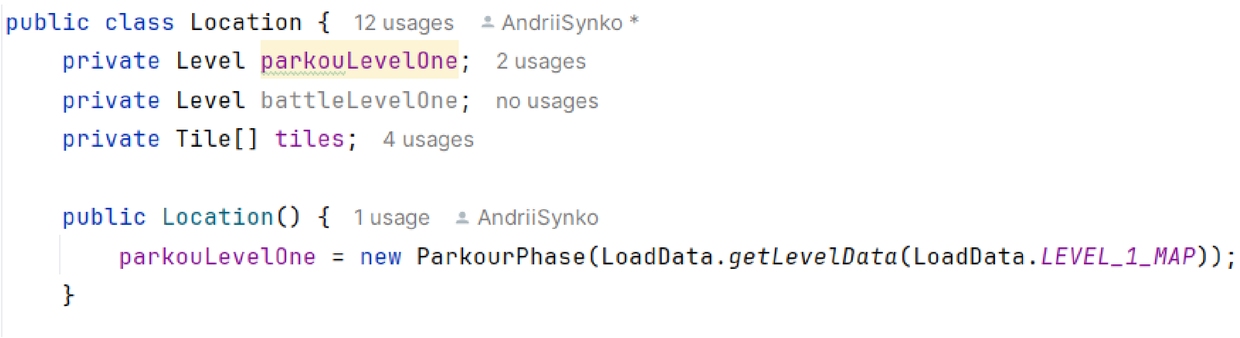
- Method overriding
  Where: src/map/ParkourPhase.java:3-9



```
public class ParkourPhase implements Level{  1 usage  AndriiSynko *
    private int[][] lvlData;  3 usages

    @Override  no usages  AndriiSynko
    public int[][] getLvlData() {
        return lvlData;
    }
```

Andrii Synko

- RunTime Polymorphism
  Where: src/map/Location.java:10-17



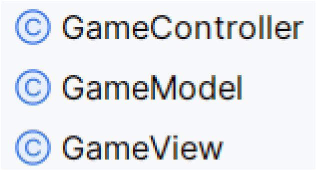
```
public class Location {  12 usages   AndriiSynko *
    private Level parkouLevelOne;  2 usages
    private Level battleLevelOne;  no usages
    private Tile[] tiles;  4 usages

    public Location() {  1 usage   AndriiSynko
        parkouLevelOne = new ParkourPhase(LoadData.getLevelData(LoadData.LEVEL_1_MAP));
    }
```

Design Patterns

MVC

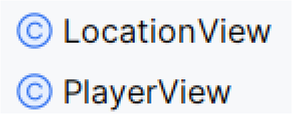
Where:

- src/gui/GameController.java (controller)
- src/gui/GameModel.java (model)
    - src/entity/Player.java (model)
    - src/map/Location.java (model)
- src/gui/GameView.java (view)
    - src/gui/PlayerView.java
    - src/gui/LocationView.java



© GameController
© GameModel
© GameView

© Location
© Player



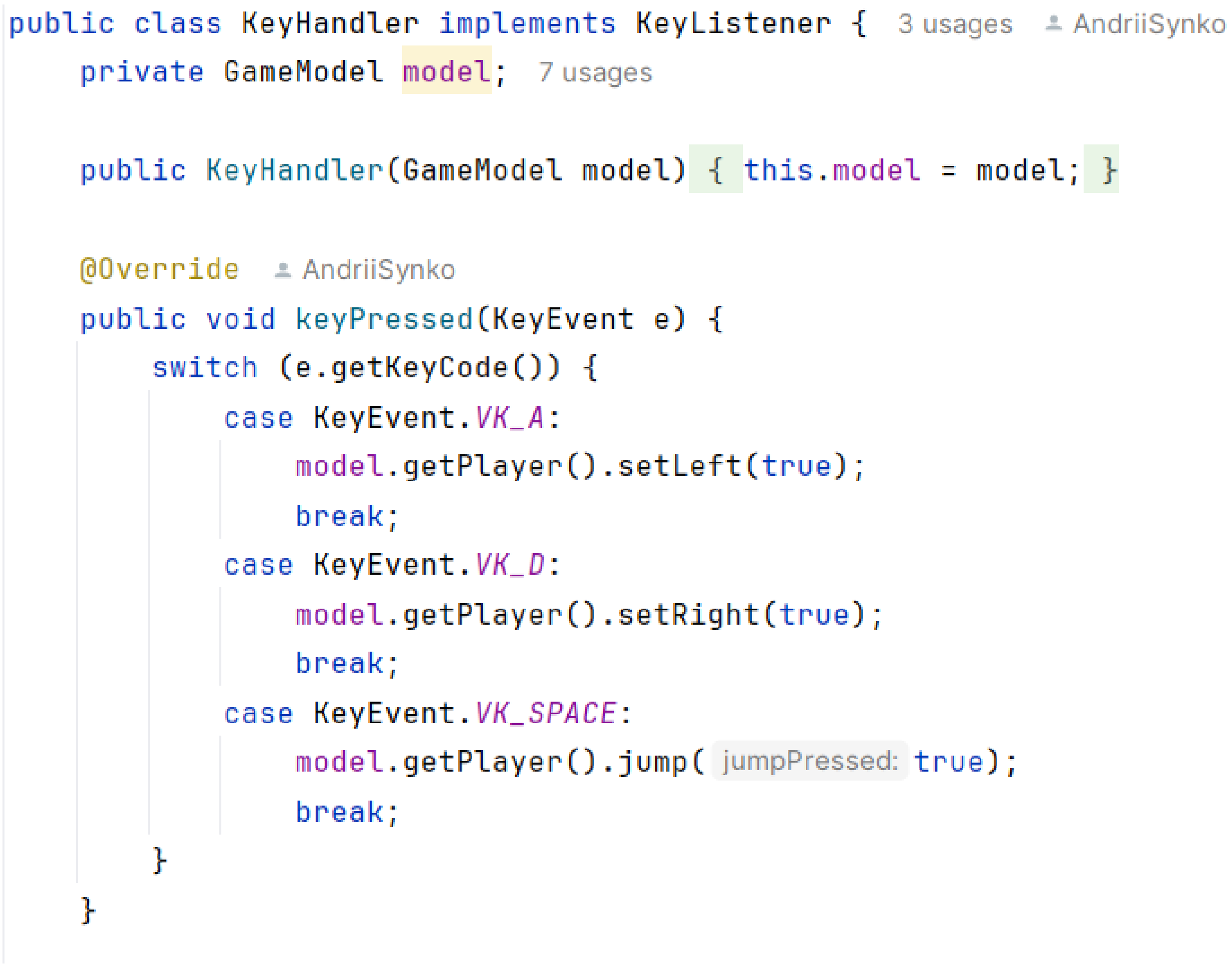
© LocationView
© PlayerView

Andrii Synko

Observer:

Where: src/input/KeyHandler.java

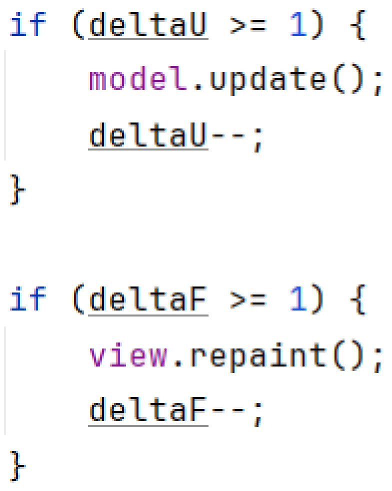
(updates player’s movement if key is pressed)



```
public class KeyHandler implements KeyListener {  3 usages  AndriiSynko
    private GameModel model;  7 usages

    public KeyHandler(GameModel model) { this.model = model; }

    @Override  AndriiSynko
    public void keyPressed(KeyEvent e) {
        switch (e.getKeyCode()) {
            case KeyEvent.VK_A:
                model.getPlayer().setLeft(true);
                break;
            case KeyEvent.VK_D:
                model.getPlayer().setRight(true);
                break;
            case KeyEvent.VK_SPACE:
                model.getPlayer().jump( jumpPressed: true);
                break;
        }
    }
```

Where: src/gameloop/GameLoop.java:39-47

(notify model to update “gamestate” and notify view to render UI after some time)



```
if (deltaU >= 1) {
    model.update();
    deltaU--;
}

if (deltaF >= 1) {
    view.repaint();
    deltaF--;
}
```