HACETTEPE UNIVERSITY

Department of Computer Science & Engineering

BBM104 Introduction to Programming Laboratory

Experiment 2

MUCAHİT VELİ CUMART – 21605893

Programming Language : Java

Subject : Inheritence and Polimorphism

Release Date : 04 /18/2019

Due Date : 05/11/2019

Teaching Assistant : Bahar GEZICI

-What was the problem?

In this assignment, we are expected to implement monopoly game with given rules, and the aim of this assignment is teaching to us the subjects,inheritence and polymorphism, in Java.

Inheritance is one of the core concepts of object-oriented programming (OOP) languages. It is a mechanism where you can to derive a class from another class for a hierarchy of classes that share a set of attributes and methods.

 Polymorphism is the ability of an object to take on many forms. The most common use of polymorphism in object oriented design occurs when a parent class reference is used to refer to a child class object.

JSON (JavaScript Object Notation) is an open-standard format that uses human-readable text to transmit data objects consisting of attributevalue pairs. It is the most common data format used for asynchronous browser/server communication, largely replacing XML which is used by AJAX.

In this Monopoly game, there are total of 40 Squres with different properties, and the player is playing the game according the dice which he/she throws. Each square has it's own properties, and according to them, he/she actions.

PROPERTY.JSON

This JSON file has the informations of Lands,Railroads and Companies, and the informations are the cost of the Property, the id of the Property and the name of the Property.

LİST.JSON

This JSON file has the informations of the Community Chest Cards and Chance Cards.

My program contains 10 class in total(Main.java,Squares.java,Lands.java,Railroads.java,Companies.java,People.java,Players.java,Banker.java,Action.java,Cards.java). One of the classes is Main class. Two of the classes are SUPERCLASSES. Six of the classes are SUBCLASSES.

SQUARES.JAVA

One of the superclasses is this class. This class is for all of the squares which the game will be played at, and this class have 4 subclass in total(Lands,Railroads,Companies,Action).

The objects of this class have two properties, id and name.

- *Lands.java:* The class is a subclass of superclass Squares.java. Has id and name from superclass and has cost and owner specially.
- *Railroads.java:* The class is a subclass of superclass Squares.java. Has id and name from superclass and has cost and owner specially.
- *Companies.java:* The class is a subclass of superclass Squares.java. Has id and name from superclass and has cost and owner specially.
- *Action.java:* The class is a subclass of superclass Squares.java. Has only id and name from superclass. This class have 3 methods. One of this methods is for action Community Chest Actions. One of the other methods is for action Chance Action,and the final method is for Cards to go under the deck.

PEOPLE.JAVA

One of the superclasses is this class. This class is for the players who will play the game (Player 1 , Player 2) and the banker. The objects of this class have two properties, id and money. And this class has 3 methods in total. One of this methods is for finding the playing player at the turn. One of this methods is for finding the player who is the other. The other method is for SHOW command. The method prints all of the wanted information to the screen.

- *Players.java:* The class is a sublcass of superclass People.java. Has id and money from superclass and has jail,jail_count,location,properties specially.
- *Banker.java:* The class is a subclass of superclass People.java. Has only id and money from superclass People.java.

CARDS.JAVA

This is the class for the cards of Community Chest and Chance. The cards have one property,and it is 'item'. We get the item, property of Cards , from LİST.JSON.

ALGORİTHM OF MY PROGRAM

- Reading the property.json and create the squares according to given information in the property.json.
- Reading the list.json and create the cards of Community Chest and Chance squares according the given information.
- Reading the commands.txt file line by line and my program executes the lines in order.
- According to the dice and if the playing player can play at the turn, the program update the location of the playing player. According to the square at the playing player updated location, perform the action and updated all properties of the players,banker and the squares.
- If there is a situation that requires the game to end or the commands in the commands.txt file is done, My program stops working.