**AP CS - Week 14 – Exercises**

**Exercise 73 - Product**

Create a class Product that represents a product sold in a webshop. A product has a price, amount and name.

A new class can be created as follows: Point at the project *073.Product* in the *projects* tab and click the right mouse button. Then select *new* and *java class*. When a dialog opens, give the class the name Product.

The class should have:

* A constructor public Product(String nameAtStart, double priceAtStart, int amountAtStart)
* A method public void printProduct() that prints a product in the following form:

Banana, price 1.1, amount 13

**Exercise 76 – Menu**

In this assignment, we will implement a class Menu that holds information about meals that are available in a cafeteria.

Our starting point is the following class skeleton:

**import** java.util.ArrayList;

**public** **class** Menu {

**private** ArrayList<String> meals;

**public** **Menu**() {

**this**.meals = **new** ArrayList<String>();

}

*// Implement the methods here*

}

Menu objects store the information of meals using an object variable of type ArrayList<String>

**Exercise 76.1: Adding a meal to menu**

Implement the method public void addMeal(String meal) that adds a new meal to the list this.meals of a Menu object. If the meal is already in the list, it should not be added.

**Exercise 76.2: Printing the menu**

Implement the method public void printMeals() that prints the meals in a menu. As an example, the output after three additions could be:

Hamburger

Fish'n'Chips

Sauerkraut

**Exercise 76.3: Clearing a menu**

Implement the method public void clearMenu() that clears a menu. After calling this method, the menu should be empty. Class ArrayList has a method that is useful here. Within your method body write meals. and see how NetBeans helps you by showing the available methods.

**Exercise 77 – Lyyra Card**

The University of Helsinki students use a so-called Lyyra cards to pay for their meals in student cafeterias. In this assignment, we implement the class LyyraCard that simulates the Lyyra card.

**Exercise 77.1: Class skeleton**

Start by adding the class LyyraCard to your project.

Then implement the LyyraCard constructor that gets the starting balance of the card as parameter. The card saves the balance in the object variable balance. Implement also the toString method that returns a string of the form "The card has X euros".

The skeleton of class LyyraCard looks like this:

**public** **class** LyyraCard {

**private** double balance;

**public** **LyyraCard**(double balanceAtStart) {

*// write code here*

}

**public** String **toString**() {

*// write code here*

}

}

The following main program can be used to test the program:

**public** **class** Main {

**public** **static** void **main**(String[] args) {

LyyraCard card = **new** **LyyraCard**(50);

System.out.**println**(card);

}

}

The output should be:

The card has 50.0 euros

**Exercise 77.2: Paying with card**

Implement the following methods to LyyraCard:

**public** void **payEconomical**() {

*// write code here*

}

**public** void **payGourmet**() {

*// write code here*

}

Method payEconomical should decrease the balance by 2.50 euros and method payGourmet by 4.00 euros.

The following main program can be used to test the program:

**public** **class** Main {

**public** **static** void **main**(String[] args) {

LyyraCard card = **new** **LyyraCard**(50);

System.out.**println**(card);

card.**payEconomical**();

System.out.**println**(card);

card.**payGourmet**();

card.**payEconomical**();

System.out.**println**(card);

}

}

The output should be:

The card has 50.0 euros

The card has 47.5 euros

The card has 41.0 euros

**Exercise 77.3: Balance nonnegative**

Change methods payEconomical and payGourmet so that if there is not enought money, the balance does not change.

The following main program can be used to test the program:

**public** **class** Main {

**public** **static** void **main**(String[] args) {

LyyraCard card = **new** **LyyraCard**(5);

System.out.**println**(card);

card.**payGourmet**();

System.out.**println**(card);

card.**payGourmet**();

System.out.**println**(card);

}

}

The output should be:

The card has 5.0 euros

The card has 1.0 euros

The card has 1.0 euros

Above, the second call payGourmet does not alter the balance since there is not enough money on the card for a gourmet lunch.

**Exercise 77.4: Loading money to card**

Add the LyyraCard the following method:

**public** void **loadMoney**(double amount) {

*// write code here*

}

The method should increase the balance of the card by the given amount. However, the maximum balance on a card is 150 euros. In case the balance after loading money would be more than that, it should be truncated to 150 euros.

The following main program can be used to test the program:

**public** **class** Main {

**public** **static** void **main**(String[] args) {

LyyraCard card = **new** **LyyraCard**(10);

System.out.**println**(card);

card.**loadMoney**(15);

System.out.**println**(card);

card.**loadMoney**(10);

System.out.**println**(card);

card.**loadMoney**(200);

System.out.**println**(card);

}

}

The output should be:

The card has 10.0 euros

The card has 25.0 euros

The card has 35.0 euros

The card has 150.0 euros

**Exercise 77.5: Loading a negative amount**

Change the method loadMoney so that the balance of the card does not change if the amount to load is negative.

The following main program can be used to test the program:

**public** **class** Main {

**public** **static** void **main**(String[] args) {

LyyraCard card = **new** **LyyraCard**(10);

System.out.**println**("Pekka: " + card);

card.**loadMoney**(-15);

System.out.**println**("Pekka: " + card);

}

}

The output should be:

Pekka: The card has 10.0 euros

Pekka: The card has 10.0 euros

**Exercise 77.6: Multiple cards**

Write a main method that does the following:

* Creates a LyyraCard for Pekka with initial balance of 20 euros
* Creates a LyyraCard for Brian with initial balance of 30 euros
* Pekka buys gourmet lunch
* Brian buys economical lunch
* cards are printed (both on their own row, starting with the name of the card owner)
* Pekka loads 20 euros
* Brian buys gourmet lunch
* cards are printed (both on their own row, starting with the name of the card owner)
* Pekka buys economical lunch
* Pekka buys economical lunch
* Brian loads 50 euros
* cards are printed (both on their own row, starting with the name of the card owner)

The main skeleton is as follows:

**public** **class** Main {

**public** **static** void **main**(String[] args) {

LyyraCard cardPekka = **new** **LyyraCard**(20);

LyyraCard cardBrian = **new** **LyyraCard**(30);

*// write code here*

}

}

The output should be:

Pekka: The card has 16.0 euros

Brian: The card has 27.5 euros

Pekka: The card has 36.0 euros

Brian: The card has 23.5 euros

Pekka: The card has 31.0 euros

Brian: The card has 73.5 euros