**Praktiskais projekts, lai pielietotu pamata OOP zināšanas.**

SR:

Aktivitāte vērsta uz šādām Programmēšana II sasniedzamajiem rezultātiem (SR):

* *Zināšanas*: Skaidro objektorientētās programmēšanas darbības pamatprincipus.
* *Prasmes*: Izmanto programmēšanas valodu un to bibliotēku dokumentāciju un palīdzības sistēmu, lai patstāvīgi apgūtu citas to piedāvātās iespējas.
* *Prasmes*: Veido programmas kādā no programmēšanas valodām, t. sk. objektorientētajā valodā.
* *Prasmes*: Veic atkļūdošanu.
* *Ieradumi*: Izmanto labās prakses principus programmas koda pierakstā un strukturēšanā.

**Saturs**

* **Uzdevuma apraksts**. Piemēram, uzņēmums “Saverma club ” vēlas vēlas automatizēt šavarmu iegādi un padarīt to ērtāku - cilvēks izvēlas šavarmas veidu, izmēru, un programmatūra visu paveic viņa vietā. Programma norāda cenu par noteikta izmēra šavarmām un izsaka personai piedāvājumu cenas veidā un jautā, vai tā piekrīt vai nē.

Darba gaita:

* **Specifikācija**
* Klases diagramma. Uzzīmēt diagrammu draw.io.
* Klases nosaukums un apraksts, kādu informāciju ietver šī klase.

import java.io.FileWriter;  
import java.util.ArrayList;  
import java.util.Scanner;  
  
public class Main {

Šīs rindas sāk galveno programmas klasi Main un tās galveno metodi main, kas ir noklusētā metode, kurā programma sāk darbību.

Scanner sc = new Scanner(System.*in*);

Izveidots jauns Scanner , kas ļauj lasīt ievadi no lietotāja.

System.*out*.println("Choose saurma type");  
System.*out*.println("1 - Kebab");  
System.*out*.println("2 - GirosPita");  
System.*out*.println("3 - Burrito");

Tiek izdrukātas ziņas konsolē, kas lūdz lietotājam izvēlēties saurma tipu.

ArrayList<Saurma> saurmas = new ArrayList<>();

**Tiek izveidots jauns ArrayList, kas saturēs saurma objektus. Saurma ir abstrakta klase, kas visiem saurma tipiem ir kopīga.**

Kebab k = new Kebab("cheddar", "pork", "+", "mayonnaise", "-", "+", "+");  
GirosPita g = new GirosPita("-", "grill chicken", "+", "mayonnaise+ketchup", "+", "+", "+",".");  
Burrito b = new Burrito("-", "chicken", "+", "barbeque", "+", "+", "+", ".");

Tiek izveidoti konkrēti saurma objekti : Kebab, GirosPita un Burrito ar noteiktām sastāvdaļām.

saurmas.add(k);

Izvēlētais Kebabs tiek pievienots sarakstam ar saurmas objektiem.

if(size==20) {  
 price = 6;  
 System.*out*.println("Price - " +price);

Ja izvēlētais izmērs ir 20, tad cena būs 6 eiro.

} else if (size==30){  
 price = 8;  
 System.*out*.println("Price - " +price);

Ja izvēlētais izmērs ir 30, tad noteikta cena būs 8 eiro.

un tā arī būs nakamās.

} else {  
 System.*out*.println("Size is not available");

Ja ir uzrakstīts kads cits izmers, tad programma nestrādas.

int o = sc.nextInt();

Deklarēšana

if(o==1){  
 System.*out*.println("Thank you!");

Ja ievadītais rezultāts ir 1, tad pasūtījums būs/ir izpildīts.

else {  
 System.*out*.println("Come again!");

Ja kaut kads cits rezultāts, tad pasūtījums nebija izpildīts.

} else if (type == 2) {  
 System.*out*.println("Chosen saurma - GirosPita");

Ja bija ievadīts cipars 2, tad cilvēks grib iegādāties girospita un talāk viss process būs tāpat kā pirma gadījuma.

Konstruktori:

1. Konstruktors inicializē Kebab objektu ar norādītajām sastāvdaļām: sieru, gaļu, tomātu, mērci, gurķi, maizi, zaļumiem.

2. Konstruktors inicializē objektu GirosPita ar norādītajām sastāvdaļām: siers, gaļa, tomāts, mērce, gurķis, vīna etiķis.

3. Konstruktors inicializē objektu Burrito ar norādītajām sastāvdaļām: siers, gaļa, tomāts, mērce, salāti, gurķis un salsa.

* Metodes: Piemēram,

printInfo() (in Kebab, GirosPita, and Burrito classes):

Šī metode atgriež virkni ar informāciju par konkrēto saurmas tipu (Kebab, GirosPita vai Burrito), ieskaitot tās sastāvdaļas.

main(String[] args) (in Main class):

Šī ir galvenā metode, ar kuru sākas programmas izpilde. Tā apstrādā lietotāja ievaddatus, lai izvēlētos saurmas veidu, lielumu un apstiprinātu pasūtījumu. Tajā ietverta arī loģika cenas aprēķināšanai, pamatojoties uz izvēlēto izmēru.

void write(String str) (in FileWriter class):

Šī metode ieraksta virkni failā, ko norāda FileWriter objekts.

void close() (in FileWriter klasē):

Šī metode aizver FileWriter objektu, atbrīvojot visus ar to saistītos sistēmas resursus.

* Padziļināti (Datu saglabāšana txt failā)

Savermas.txt ir visa informācija,kuru programma piedāvāja lietotājam un secināja pasūtījumu.

* **Secinājumi**. (Ieteikumi programmas uzlabošanai. Kas izdevās/neizdevās?)

Man liekas, ka izdevās viss, tikai varēja pievienot vairāk šavarmas veidu, bet diemžēl vairs nav.

* **Pielikums**. Nokopēt kodu pielikumā ar rindu numerāciju vai saite uz kodu.