|  |  |  |
| --- | --- | --- |
| American University of SharjahSchool of Engineering Department of Computer Engineering  P. O. Box 26666  Sharjah, UAE  Failure to put your name and ID will result in a 2-point deduction from you grade.  **Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **ID : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |  | **Instructor:** Imran A. Zualkernan **Office**: ESB-2066  **Phone**: 971-6-515 2953  **Fax**: 971-6-515 2979  **e-mail**: izualkernan@aus.edu  **Semester**: Fall 2020 |

COE312 – Quiz 2 – Solution

Time: 45 minutes

Total Points: 10

Solution

**The quiz open book and notes and you can use the Internet. You are not allowed to consult other individuals or your classmates.**

PLEASE UPLOAD YOUR ANSWER USING THIS **WORD** FILE. DO NOT UPLOAD ANY ZIPPED OR PDF FILE. ONE WORD FILE ONLY.

The question(s) have been written in a manner such that it is not possible for two students to have the same solution. Therefore, please refrain from the temptation of copying code and changing order of and names of variables, etc. AUS code of conduct will be strictly enforced, and no violation will be tolerated as per AUS policy. Please note that the AUS code does not discriminate between who copied from whom so it is not advisable to share your solution.



**Q1.** (10 points) Develop a Java class hierarchy to model a karak tea shop that serves ONLY karak tea. You must use all relevant Design patterns in your code.

The karak tea is made from:

* Milk (whole, condensed or evaporated milk)
* Water
* Sugar
* Spices – Cardamom, Ginger and Cinnamon
* A strong black tea base

Each tea can have the following optional ingredients.

* A whole clove
* Pepper
* Star anise
* Honey
* Saffron

The karak tea shop can have multiple customers who can come and go. A customer can order a tea and then wait for the delivery of their tea. Assume that each customer can only order one tea at a time. The teashop has many waiters to take the orders and one cook for making the tea.

You must run and provide output on the following program (without any changes to the program).

**import** java.util.Random;

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

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

**final** **int** num\_waiters = 5;

Random r = **new** Random();

//create a teashop with waiters and one cook

TeaShop ts = **new** TeaShop(num\_waiters);

//create 10 customers placing orders with random

//waiters of the shop

Customer c;

**for**(**int** i = 0; i< 10; i++) {

Thread.*sleep*(r.nextInt(5000)); // within the next 5 seconds

c = **new** Customer(ts);

c.place\_order(r.nextInt(num\_waiters), **new** Karak(c));

}

}

}

**Expected Output:**

Waiter@5a07e868 received an order from the customer Customer@76ed5528

Cook@2c7b84de is making the tea Karak@3fee733d on Fri Nov 20 14:42:57 GST 2020

Customer@76ed5528 received their tea Karak@3fee733d on Fri Nov 20 14:42:58 GST 2020

Waiter@3f99bd52 received an order from the customer Customer@4f023edb

Cook@2c7b84de is making the tea Karak@3a71f4dd on Fri Nov 20 14:43:02 GST 2020

Customer@4f023edb received their tea Karak@3a71f4dd on Fri Nov 20 14:43:02 GST 2020

Waiter@5a07e868 received an order from the customer Customer@7adf9f5f

Cook@2c7b84de is making the tea Karak@85ede7b on Fri Nov 20 14:43:03 GST 2020

Customer@7adf9f5f received their tea Karak@85ede7b on Fri Nov 20 14:43:06 GST 2020

Waiter@5674cd4d received an order from the customer Customer@63961c42

Cook@2c7b84de is making the tea Karak@65b54208 on Fri Nov 20 14:43:10 GST 2020

Customer@63961c42 received their tea Karak@65b54208 on Fri Nov 20 14:43:12 GST 2020

Waiter@1be6f5c3 received an order from the customer Customer@6b884d57

Cook@2c7b84de is making the tea Karak@38af3868 on Fri Nov 20 14:43:14 GST 2020

Customer@6b884d57 received their tea Karak@38af3868 on Fri Nov 20 14:43:16 GST 2020

Waiter@5a07e868 received an order from the customer Customer@77459877

Cook@2c7b84de is making the tea Karak@5b2133b1 on Fri Nov 20 14:43:18 GST 2020

Customer@77459877 received their tea Karak@5b2133b1 on Fri Nov 20 14:43:18 GST 2020

Waiter@5a07e868 received an order from the customer Customer@72ea2f77

Cook@2c7b84de is making the tea Karak@33c7353a on Fri Nov 20 14:43:21 GST 2020

Customer@72ea2f77 received their tea Karak@33c7353a on Fri Nov 20 14:43:22 GST 2020

Waiter@1be6f5c3 received an order from the customer Customer@681a9515

Cook@2c7b84de is making the tea Karak@3af49f1c on Fri Nov 20 14:43:25 GST 2020

Customer@681a9515 received their tea Karak@3af49f1c on Fri Nov 20 14:43:26 GST 2020

Waiter@3f99bd52 received an order from the customer Customer@19469ea2

Cook@2c7b84de is making the tea Karak@13221655 on Fri Nov 20 14:43:26 GST 2020

Customer@19469ea2 received their tea Karak@13221655 on Fri Nov 20 14:43:26 GST 2020

Waiter@5a07e868 received an order from the customer Customer@2f2c9b19

Cook@2c7b84de is making the tea Karak@31befd9f on Fri Nov 20 14:43:28 GST 2020

Customer@2f2c9b19 received their tea Karak@31befd9f on Fri Nov 20 14:43:30 GST 2020

Please provide 1) formatted code, and 2) screenshots of your running program. Not providing a screenshot (with or without errors) will limit your score to below 3/10.

**Grading Rubric (out of 10 points)**

|  |  |  |  |
| --- | --- | --- | --- |
| **0-3** | **4-6** | **7-8** | **9-10** |
| Program does not compile or run. OR  Program is not related to the problem at hand. OR  No screenshot is provided. OR  No design patterns are used. | * Class hierarchy is correct. AND * Most interfaces are implemented correctly. AND * Some attributes are missing. * At least one design pattern is used correctly. | * Class hierarchy is correct. AND * Most interfaces are implemented correctly. AND * Some obvious attributes are missing. * Some design patterns are implemented but not entirely correctly. * Some design patterns are used unnecessarily. | * Class hierarchy is correct. AND * All the interfaces are implemented correctly. AND * All the attributes are mentioned. AND * All design patterns are used appropriately, * Proper encapsulation principles are followed. * No design pattern is used if not required. |

**Solution: (Describe the idea behind your solution)**

**Solution: (Paste formatted code here).**

**Solution: (Paste the screenshots here).**