**COMSATS University Islamabad**

Sahiwal Campus



**CSC241 Object Oriented Programming**

Experiment 02: Object Oriented Modeling & Control Structures

DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

Prepared By: Checked By: Approved By:

Engr. M. Hamid Saeed Ma’am Sadia Bano OBE committee

**----------------------------------- ----------------------------------- -----------------------------------**

# Lab Experiment 02:

### Object Oriented Modeling & Control Structures

**Objectives:**

1. To explore the concept behind Object Oriented Modeling
   * Identify entities in a simple ―personal banking‖ application domain.
   * Identify attributes required to model these entities.
   * Identify methods required to model these entities.
   * Identify relationships among these entities.
2. Implementation of Control Structures
   * Selection, and Iterative statements

**Pre-Lab**

**A Case Study:**

1. **The Personal Banking Domain**

In this exercise the students will investigate the different kinds of objects required to model the banking domain. To make it simple, the scope of the model is limited to personal banking activities, i.e. excluding business banking.

1. **Brainstorming**

Consider a world in which people can save money into a bank. There may be different banks (e.g. Bank Al-Habib, Allied Bank etc) and customers may have different activities with their banks.

* Write down the different kinds of objects which are involved in this ―Personal banking‖ domain.
* Look at the objects that you put down above. Is/are there any object(s) that you chose not to include in your model? What are they? Why?

1. **Customer**

Depending on your brain-storming result in exercise above, you may have some Customer objects. Consider a customer who can open bank accounts in a bank. Write down the list of attributes required to model the data components of a customer, (e.g. the name of the customer) For each of these attributes:

* Would you model it as a class-level/object-level attribute and why?
* What will be the appropriate data type? (e.g. integer, floating point number, a string, etc.)
* Would you make this attribute visible to other objects and why?
* Would you hide this attribute from other objects and why?
* Write down the list of methods required to model the procedural component of a customer, i.e. Operations/ behavior that a customer can perform. For each of these methods:
* Would you model it as a class-level/object-level method and why?
* Does the method need any input argument/ parameter? If yes, what is/are the data type(s) of this/these parameter(s)?
* Does the method return any value? If yes, what is the appropriate return data type?
* Would you make the method visible/invisible to other objects? Why?

1. **Bank Account**

A bank account stores data about the amount of money that a customer saves in a bank. Write down the list of attributes required to model a bank account. For each of these attributes:

* Should it be object/class-level and why?
* What is the appropriate data type? (E.g. integer, floating point number, a string, etc.)
* Would you make the attribute visible to the outside world and why?
* Write down the list of methods required to model a bank account. For each of these methods:
* Should it be object/class-level and why?
* Does the method need any input argument/ parameter? If yes, what is/are the data type(s) of this/these parameter(s)?
* Does the method return any value? If yes, what is the appropriate return data type?
* Would you make the method visible/invisible to other objects? Why?

Consider the relationship between a bank account and a customer.

* Do you think this is a one-to-one, one-to-many, or many-to-many relationship? Why?
* If you need to model this relationship between a bank account and a customer, would you model it as an attribute or method? Why and How?
* Would you make it an attribute/method of the bank account or the customer?
* What would be the data type of this attribute/method?

**5. The Bank**

A bank account stores data about the amount of money that a customer saves in a bank. Write down the list of attributes and methods required to model a Bank. For each of this attributes/method:

* Would you model it as a class/object-level attribute/method? Why?

For each attribute:

* What is the appropriate data type?
* For each method:
* What is/are the required input argument(s)/parameter(s)?
* What is/are its/their data type(s)?
* What is the return data type (if any)?
* Consider the bank and bank accounts:
* Is there any relationship between a bank and bank accounts?
* Is it a one-to-one, one-to-many, or many-to-many relationship?
* How would you model this relationship?
* Consider the bank and customer:
* Is there any relationship between banks a customer?

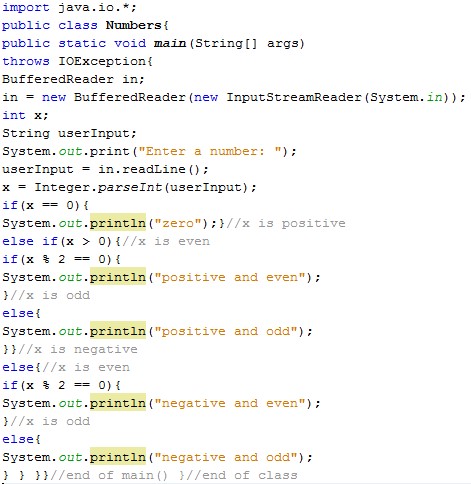
Will you model this relationship? If yes, how? If you are going to have multiple banks in our domain (e.g. Bank Al-Habib, Allied Bank etc.), will it make a difference to your domain model? If you need to make any change, how will you modify it?

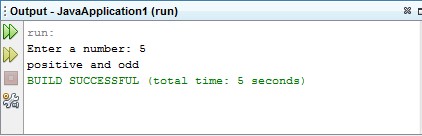
**In-Lab**

**Control Structures**

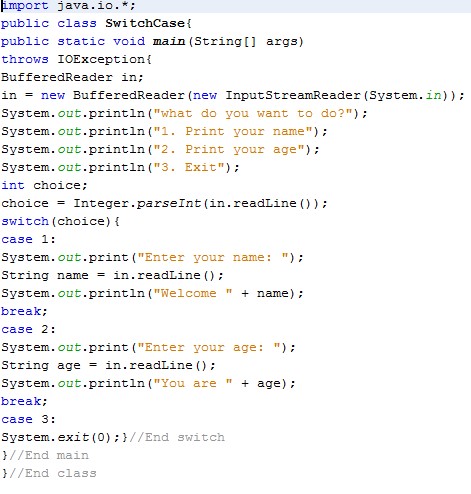
Sometimes we need to decide between two choices. The statements in by which we make this decision are called conditional statements. We use loops to make one or more statements repeat for specified number of times. Clearly this have an importance when we wish to take more than one input from user or do some calculations.

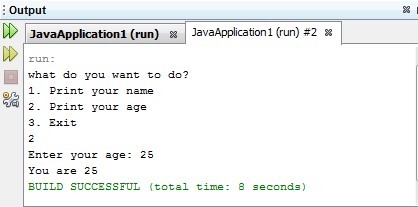
Example 1: This program is used to demonstrate the uses if-else condition.



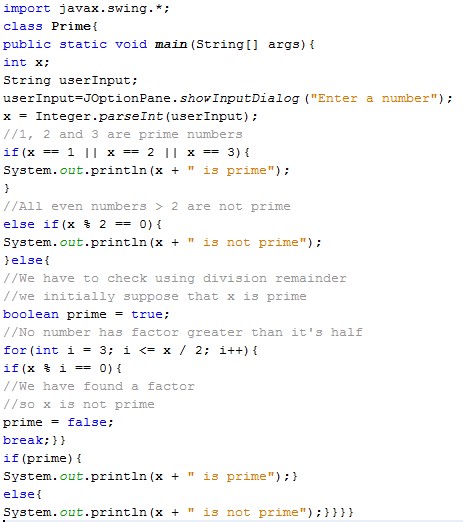


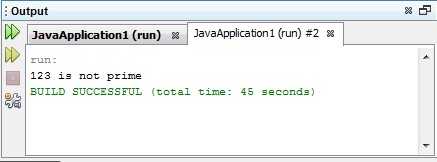
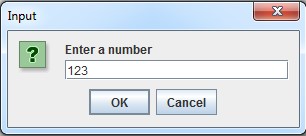
Example 2: This program is used to demonstrate the uses switch-case.



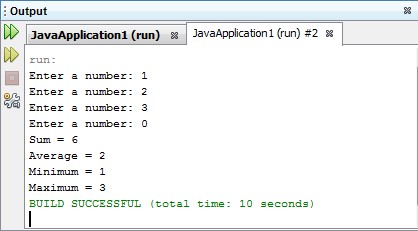
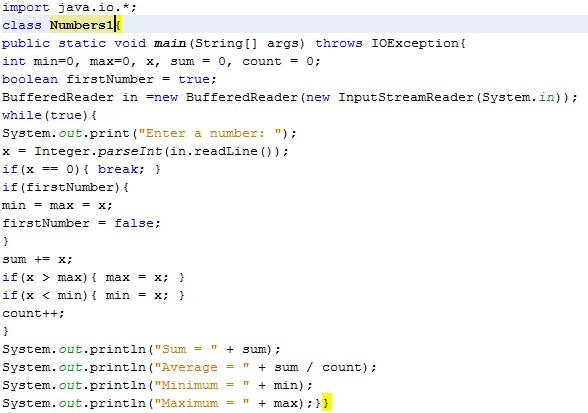


Example 3:This program demonstrates either an input number is prime or not





Example 4:Following program that keeps reading numbers (integers) from user until input is 0, then prints average, sum, maximum and minimum.



**Exercise Task 1:**

Write a program that specifies whether a given number (x) falls in one of the following categories (give x a value from the code, don't read from user):

* 0 to 9
* 10 to 19
* 20 to 29
* None of the categories

For example, if x = 5, program should print "0 to 10", and if x = 44 it should print "None". Hint: use if statements with Boolean expressions combined using &&.

**Exercise Task 2:**

Write a program that reads two integers and prints their sum like the code below (text shown in boldface is supposed to be user input). Enter the first number: 3

Enter the second number: 4

The sum is 7

**Exercise Task 3:**

Write a program that asks the user to enter two numerical values (integers) and then select an operation (addition, subtraction, multiplication and division) then prints the result based on operation selected. The code below shows examples of the output (text shown in boldface is supposed to be user input). Enter first number: 4 Enter second number: 2 1. Addition (+).

1. Subtraction (-).
2. Multiplication (\*).
3. Division (/).

Enter operation number: 3

The result is 8

**Exercise Task 4:**

Write a program that reads 10 numbers from the user then prints out how many positive numbers and negative numbers user has entered (consider 0 a positive number).

## Post Lab

**Q.1** Modify calculation program in exercise task 3 by adding the following question at the end of the program:

Do you want to make another calculation?

* 1. Yes
  2. No

Enter your option:

If user selects yes (by entering 1), program will ask him again to enter new two numbers and select operation, if user selects no (by entering 2), program exits. Use appropriate loop to accomplish this.

**Q.2** Write a program that asks the user to enter certain number, after that asks him to enter another 20 numbers, after entering them all, it prints out the number of occurrences of the first number. See the below example (text shown in boldface is supposed to be user input).

Enter number to search for: 2

Enter a number: (20 times)

2 3 56 7 9 2 4 5 5 6 2 21 33 19 32 88 0 32 100 20 The number (2) occurred 3 times in your input.

**Note:** Submit your Lab report consisting of all in-lab exercise tasks results and post lab tasks within two working days. After the deadline your lab report won’t be considered.