Name : Ruby.S

Batch : 2022 – 7607

Enrolment number : EBEON0722634378

Java Introductions

Java is a high-level programming language Java is a platform independent language. it runs on a multiple platforms such as UNIX, Windows, Mac OS. The Java language’s programming is based on the concept of OOP.

Java is a case sensitive language
Ex: NAME and name are not same as per Java Language
2. Java file name should be same as its Class name
3. Class name should start with upper case letter
4. Method name should start with lower case letter
5. Every statement should end with semi colon
6. Java program execution starts from main method which is
mandatory in every program

JDK, JRE AND JVM

We need to understand three terminologies for sure in Java such as JDK, JRE and JVM.

JDK

JDK stands for Java Development Kit.

Using JDK, we can develop, compile and execute (run) new applications and also we can modify existing applications.

JRE

JRE stands for Java Runtime Environment.Using JRE, we can only execute already developed applications. We cannot develop new applications or modify existing applications.

JVM

JVM stands for Java Virtual Machine. JVM drives the java code. Using JVM, we can run java byte code by converting them into current OS machine language.

Comments.

In Java, we have two types of comments. We use comments to write some text within our code. Compiler will ignore these comments

Variable.

In Java, variable is a name given to a memory location and this variable is associated with a value.

There are three types of variables in Java.

1. Local variable
2. Instance variable
3. Class/Static variable

Data types

Data types in java specify the size and type of values that can be stored in an identifier. There are two types of Data Types in Java.

1. Primitive Data Type
2. Non-primitive Data Type

Primitive Data Type

There are 8 primitive data types such as byte, short, int, long, float, double, char, and boolean.

Non-primitive Data Type

Non-primitive data types include Classes, Interfaces and Arrays

Operators

Operators in Java are the special symbols that perform specific operations and then return a result.

Types of Operators in Java are

1. Arithmetic Operators (+, –, *****, /, **%**)
2. Assignment Operators (=, +=, -=, *****=, /=, **%**=)
3. Auto-increment Operator and Auto-decrement Operators (++, —)
4. Logical Operators (**&&**, ||, **!**)
5. Comparison (relational) Operators (==, **!**=, >, <, >=, <=)
6. Bitwise Operators (**&**, |, ^, ~, <<, >>**)**
7. Ternary Operator

conditional statements

Let’s see the following conditional statements

1.if statement
2.nested if statement
3.if-else statement
4.if-else-if statement
5.Switch Case Statement

Continue Statement,Break statement

OOPS concept

OOPS Stands for Object Oriented Programming System. In this tutorial, I will introduce you to Class, Object, Constructor, Abstraction, Encapsulation, Inheritance, Polymorphism, Interface etc.,

Class

A class is a blueprint or prototype from which objects are created. A class contains variables (data types) and methods (functions) to describe the behavior of an object.

Object

Object is a software bundle of related state and behavior. Objects have two characteristics namely state and behavior.

Object is an entity that has state and behavior.

State: It represents value (data types/variables) of an object
Behavior: It represents the functionality (methods) of an object

Object is an instance of a class.

Methods

Methods are also known as procedures or functions

Methods are of two types

1. Access Modifiers: Access modifiers are subdivided into four types such as Default, Public, Private, Protected
2. Non-access Modifiers: Non-access modifiers are subdivided into four types such as Static, Final, Abstract, Synchronized

Constructor

Constructor name should be same as class name. It looks like a method but its not a method. It wont return any value.

Inheritance

Inheritance is a process where one class inherits the properties of another class.

Polymorphism

Polymorphism allows us to perform a task in multiple ways. Let’s break the word Polymorphism and see it, ‘Poly’ means ‘Many’ and ‘Morphos’ means ‘Shapes’.

There are three ways to overload a method.

1. Parameters with different data types
2. Parameters with different sequence of a data types
3. Different number of parameters overriding means to override the functionality of an existing method.

Abstraction

Abstraction is the methodology of hiding the implementation of internal details and showing the functionality to the users.

Encapsulation

Encapsulation is a mechanism of binding code and data together in a single unit. Let’s take an example of Capsule.

Array

Collection of similar type of elements is known as Array. Array in Java is an Object that holds fixed number of values of a similar data types which means an array of int will contain only integers, an array of string will contain only strings etc..

The Java programming language uses exceptions to handle errors and other exceptional events.

1. Checked Exceptions
2. Unchecked Exceptions

MY PROJECT IS

ATM MACHINE

An automated teller machine (ATM) is **a specialized computer that allows you to complete bank transactions without the need to see a bank representative**.

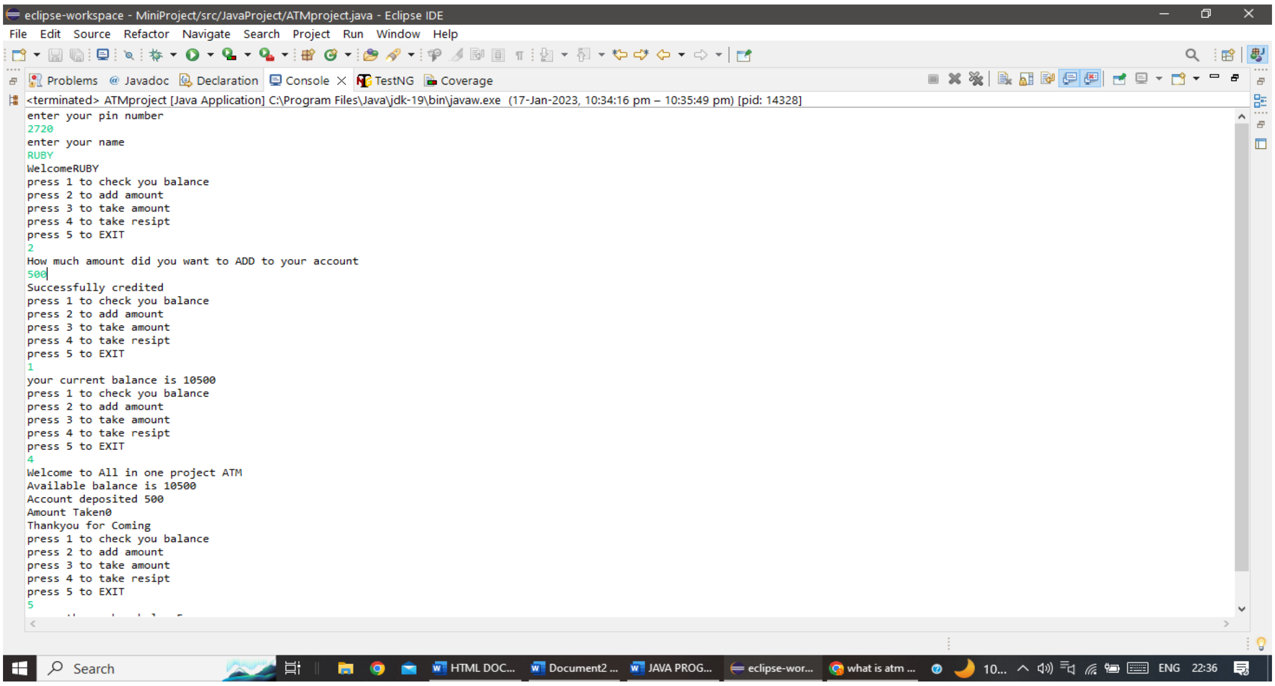
JAVA CODING

```
package JavaProject;

import java.util.Scanner;

public class ATMproject {

public static void main(String[] args) {

int pin=2720;

int balance=10000;

int AddAmount=0;

int TakeAmount=0;

String name;

Scanner scanner=new Scanner(System.in);

System.out.println("enter your pin number");

int password=scanner.nextInt();

if(password==pin)

{

System.out.println("enter your name");

name=scanner.next();

System.out.println("Welcome"+name);

while (true)

{

System.out.println("press 1 to check you balance");
System.out.println("press 2 to add amount");

System.out.println("press 3 to take amount");

System.out.println("press 4 to take resipt");

System.out.println("press 5 to EXIT");

int opt=scanner.nextInt();

switch(opt)

{

case 1:

System.out.println("your current balance is " + balance);

break;

case 2:

System.out.println("How much amount did you want to ADD to your
account");

AddAmount= scanner.nextInt();

System.out.println("Successfully credited");

balance=AddAmount + balance;

//10000=100+10000

//balance=10100

break;

case 3:

System.out.println("How much amount did you want to take");

TakeAmount=scanner.nextInt();
```

```
if(TakeAmount>balance)

{

System.out.println("your bank balance is insufficient");

System.out.println("try less than your available balance");

}

else

{

System.out.println("Successfully taken");

balance=balance-TakeAmount;

//balance=10100=10100-100

//balance=10000

}

break;

case 4:

System.out.println("Welcome to All in one project ATM");

System.out.println("Available balance is "+ balance);

System.out.println("Account deposited "+AddAmount);

System.out.println("Amount Taken"+TakeAmount);

System.out.println("Thankyou for Coming");

break;

default:
```

```
System.out.println("press the number below 5");

break;

}

if(opt==5)

{

System.out.println("thankyou");

break;

}

}

}

else

{

System.out.println("Wrong pin number");

}

}

}
```

OUTPUT AND SCREENSHOT



eclipse-workspace - MiniProject/src/JavaProject/ATMproject.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Problems
Javadoc
Declaration
Console
TestNG
Coverage
Search
HTML DOC...
Document2 ...
JAVA PROG...
eclipse-wor...
what is atm ...
ENG
22:36

```
<terminated> ATMproject [Java Application] C:\Program Files\Java\jdk-19\bin\javaw.exe  (17-Jan-2023, 10:34:16 pm – 10:35:49 pm) [pid: 14328]
enter your pin number
2720
enter your name
RUBY
WelcomeRUBY
press 1 to check you balance
press 2 to add amount
press 3 to take amount
press 4 to take resipt
press 5 to EXIT
2
How much amount did you want to ADD to your account
500
Successfully credited
press 1 to check you balance
press 2 to add amount
press 3 to take amount
press 4 to take resipt
press 5 to EXIT
1
your current balance is 10500
press 1 to check you balance
press 2 to add amount
press 3 to take amount
press 4 to take resipt
press 5 to EXIT
4
Welcome to All in one project ATM
Available balance is 10500
Account deposited 500
Amount Taken0
Thankyou for Coming
press 1 to check you balance
press 2 to add amount
press 3 to take amount
press 4 to take resipt
press 5 to EXIT
5
```

In this screenshot first the machine will ask your pin number

After we entered it will ask your name

After entered name it will display

Welcome RUBY

press 1 to check you balance

press 2 to add amount

press 3 to take amount

press 4 to take receipt

press 5 to EXIT

we can press any number to check your details.

COLLECTIONS

The **Collection in Java** is a framework that provides an architecture to store and manipulate the group of objects.

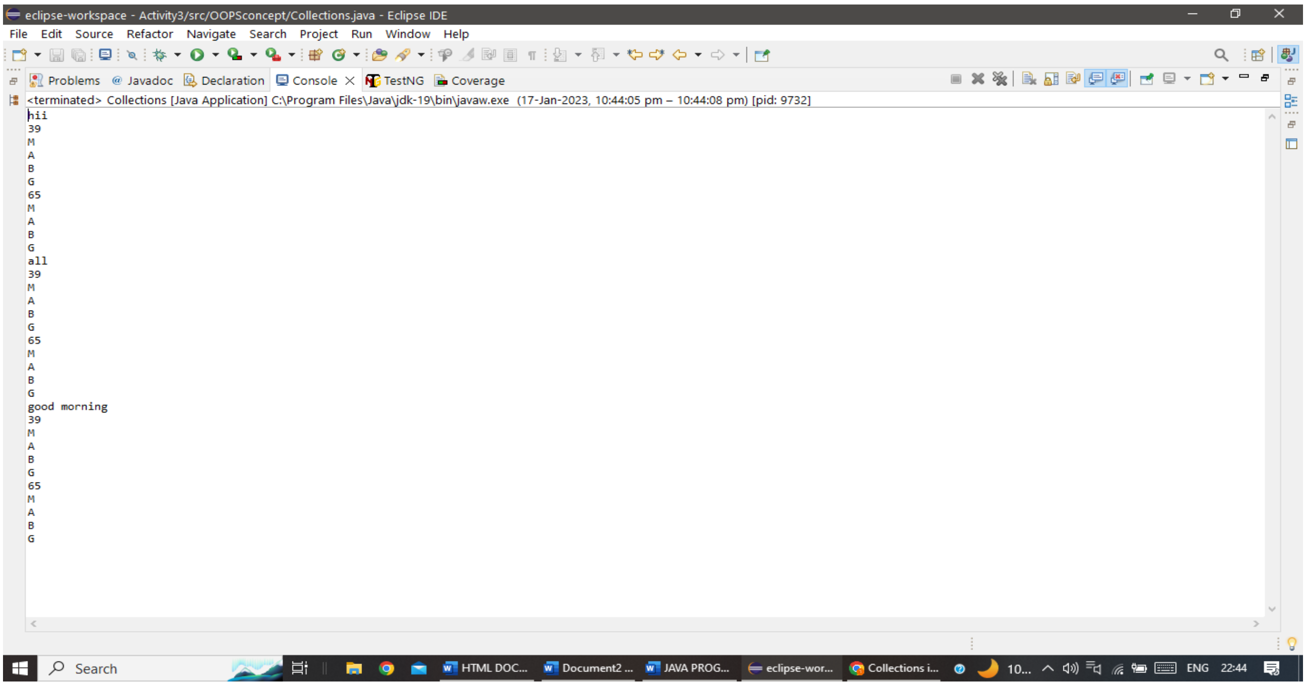
Java Collections can achieve all the operations that you perform on a data such as searching, sorting, insertion, manipulation, and deletion.

Java Collection means a single unit of objects. Java Collection framework provides many interfaces (Set, List, Queue, Deque) and classes (ArrayList, Vector, LinkedList, PriorityQueue, HashSet, LinkedHashSet, TreeSet).

SAMPLE CODING

```
package OOPSconcept;
import java.util.*;
public class Collections {
public static void main(String[] args) {
Deque<String> deque = new ArrayDeque<String>();
deque.add("hii");
deque.add("all");
deque.add("good morning");
for (String st : deque) {
System.out.println(st);
HashSet<String> set=new HashSet<String>();
set.add("39");

set.add("65");

set.add("65");

set.add("39");

Iterator<String> itr=set.iterator();

while(itr.hasNext()){

System.out.println(itr.next());

List<String> list=new ArrayList<String>();

list.add("M");

list.add("A");

list.add("B");

list.add("G");

for(String letters:list)

System.out.println(letters);

}

}

}

}
```

OUTPUT AND SCREENSHOT



eclipse-workspace - Activity3/src/OOPSconcept/Collections.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Problems
Javadoc
Declaration
Console
TestNG
Coverage
Search
HTML DOC...
Document2 ...
JAVA PROG...
eclipse-wor...
Collections i...
10...
ENG
22:44

```
<terminated> Collections [Java Application] C:\Program Files\Java\jdk-19\bin\javaw.exe  (17-Jan-2023, 10:44:05 pm – 10:44:08 pm) [pid: 9732]
hii
39
M
A
B
G
65
M
A
B
G
all
39
M
A
B
G
65
M
A
B
G
good morning
39
M
A
B
G
65
M
A
B
G
```

In this output we can see the collection program result.