ASSIGMENT JAVA DAY17

Harshit Kushmakar| 16896

1. **Create a JAR file of the BRD 1 – Validation Functions and install it in the .m2 repository. Use the same as a dependency in another Project and call the Validation methods to test the inclusion.**

```
package com.validation.kushmakar;

import java.time.LocalDate;
import java.time.format.DateTimeFormatter;
import java.util.List;

public class Validation {
    public boolean dataTypeValidation(Object fieldValue, String dataType) {
        if(dataType.equals(fieldValue.getClass().getSimpleName())){
            return true;
        }
        return false;
    }

    public boolean dataLengthValidation(Object value, int maxLength){
        if(value.toString().length()<=maxLength){
            return true;
        }
        return false;
    }

    public boolean specialCharacterValidation(String value, String
specialChar) {
        for (int i = 0; i < specialChar.length(); i++) {
            if (value.contains(specialChar.charAt(i) + "")) {
                return true;
            }
        }
        return false;
    }

    public <T> boolean domainValueValidation(Object value, List<T> domain){
        for(T t:domain){
            if(value.equals(t)){
                return true;
            }
        }
        return false;
    }


    public boolean formatValidation(LocalDate value, String format){
        DateTimeFormatter formatObj = DateTimeFormatter.ofPattern(format);
        String newValue = value.format(formatObj);
        if(newValue.equals(value.toString())){
```

```
            return true;
        }
        return false;
    }


    public boolean validateEmail(String email)
    {
        if(email.contains("@")&& email.contains(".com")&&!email.contains("
")){
            return true;
        }
        return false;
    }

}
```

main:

```
package com.validation.kushmakar;

import java.time.LocalDate;
import java.util.ArrayList;
import java.util.List;

public class ValidationMain {
    public static void main(String[] args) {
        int a = 90;

        Validation validation = new Validation();
        System.out.println(validation.dataTypeValidation(a, "Integer"));
        System.out.println(validation.dataLengthValidation(100000, 1));
        System.out.println(validation.specialCharacterValidation("hhhj",
"#"));
        List<Integer> domainList = new ArrayList<>();
        domainList.add(5);
        domainList.add(6);
        domainList.add(7);

        System.out.println(validation.domainValueValidation(7,
domainList));
        System.out.println(LocalDate.now());
        System.out.println(validation.formatValidation(LocalDate.now(),
"YYYY-MM-DD"));
        System.out.println(validation.validateEmail("abc@email.com"));


    }
}
```

OUTPUT

```
true
false
false
true
2023-02-12
false
true

Process finished with exit code 0
```

Created a tester class and Added dependency of Brd-validation in pom.xml and build it.

pom.xml (tester-class) | Main.java

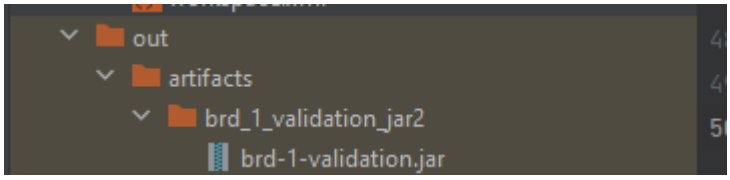
```xml
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>

    <groupId>org.example</groupId>
    <artifactId>tester-class</artifactId>
    <version>1.0-SNAPSHOT</version>

    <properties>
        <maven.compiler.source>11</maven.compiler.source>
        <maven.compiler.target>11</maven.compiler.target>
        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    </properties>
<dependencies>
    <dependency>
        <groupId>com.validation.kushmakar</groupId>
        <artifactId>brd-1-validation</artifactId>
        <version>1.0-SNAPSHOT</version>
    </dependency>

</dependencies>
</project>
```

```java
package org.example;

import com.validation.kushmakar.Validation;

import java.time.LocalDate;
import java.util.ArrayList;
import java.util.List;

public class Main {
    public static void main(String[] args) {


            int a = 90;

            Validation validation = new Validation();
            System.out.println(validation.dataTypeValidation(a, dataType: "Integer"));
            System.out.println(validation.dataLengthValidation( value: 100000, maxLength: 1));
            System.out.println(validation.specialCharacterValidation( value: "hhhj", specialChar: "#"));
            List<Integer> domainList = new ArrayList<>();
            domainList.add(5);
            domainList.add(6);
            domainList.add(7);

            System.out.println(validation.domainValueValidation( value: 7, domainList));
            System.out.println(LocalDate.now());
            System.out.println(validation.formatValidation(LocalDate.now(), format: "YYYY-MM-DD"));
            System.out.println(validation.validateEmail("abc@email.com"));
```

output(created a tester class in the new project to call the methods**).**

```
"C:\Program Files\Java\jdk-11\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2022.2.3\lib\idea_rt.jar=65417:C:\Program
true
false
false
true
2023-02-12
false
true

Process finished with exit code 0
```

.jar file of BRD-Validation.



2. Given below is the class diagram for the classes already created during the Inheritance and Collections assignments. Reuse the same classes. Create multi-module project for the above diagram as mentioned below:

1. Take Bank class in module1 and rest of the classes in module2.

2. Main class is in module1.

Test the functionalities.

Module 1:

Bank class

```
package org.example;

public class Bank extends Customer {



        private Customer[] customers = new Customer[1000];
        private int size = 0;

        public void setCustomers(Customer[] customers) {
            this.customers = customers;
        }

        public Customer[] getCustomers() {
            return customers;
        }

        public boolean registerCustomer(Customer c) {
            if (c == null) {
                return false;
            }
            customers[size++] = c;
            return true;
        }

        public boolean deleteCustomer(Customer customerId) {
            return false;
        }
```

```
        public boolean findCustomer(Customer customer) {
            for (Customer c : customers) {
                if (c.equals(customer)) {
                    return true;
                }
            }
            return false;
        }

        public void printAllCustomers() {
            for (Customer c : customers) {
                System.out.println(c.toString());
            }
        }

        public boolean deleteCustomer(int customerId) {
            int check = 0;
            for (int i = 0; i < customers.length; i++) {
                if (customers[i].getCustomerId() == customerId) {
                    check = 1;
                    break;
                }
            }
            if (check == 0) {
                return false;
            }
            for (int i = 0; i < customers.length; i++) {
                customers[i] = customers[i + 1];
            }
            return true;
        }

        public boolean findCustomer(String customer) {
            return false;
        }



    }
```

Main

```
package org.example;

public class Main {
    public static void main(String[] args) {
       Bank bank = new Bank();
        System.out.println(bank.findCustomer("aman"));
    }
}
```

pom.xml

pom.xml (Module1) | Bank.java | Main.java

```xml
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>

    <groupId>org.example</groupId>
    <artifactId>Module1</artifactId>
    <version>1.0-SNAPSHOT</version>

    <properties>
        <maven.compiler.source>11</maven.compiler.source>
        <maven.compiler.target>11</maven.compiler.target>
        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    </properties>

    <dependencies>
        <dependency>
            <groupId>org.example</groupId>
            <artifactId>Module2</artifactId>
            <version>1.0-SNAPSHOT</version>
        </dependency>
    </dependencies>

</project>
```

Module 2:

LoanAgreement:

```java
package org.example;

import java.time.LocalDate;

public class LoanAgreement{

    private LocalDate loanDisbursalDate;

    private double income;
    public LoanAgreement( String loanDisburseDate, double salary){

        this.loanDisbursalDate =  LocalDate.parse(loanDisburseDate);
        this.income = salary;

    }


    private int loanAgreementId;
    private class LoanProduct{}
    private double loanAmount;
```

```
    private int tenure;
    private double roi;
    private enum loanStatus{}
    private double emiPerMonth;
    //    private LocalDate loanDisbursalDate;
    private int repaymentFrequecy;

    public int getLoanAgreementId() {
        return loanAgreementId;
    }

    public void setLoanAgreementId(int loanAgreementId) {
        this.loanAgreementId = loanAgreementId;
    }

    public double getLoanAmount() {
        return loanAmount;
    }

    public void setLoanAmount(double loanAmount) {
        this.loanAmount = loanAmount;
    }

    public int getTenure() {
        return tenure;
    }

    public void setTenure(int tenure) {
        this.tenure = tenure;
    }

    public double getRoi() {
        return roi;
    }

    public void setRoi(double roi) {
        this.roi = roi;
    }

    public double getEmiPerMonth() {
        return emiPerMonth;
    }

    public void setEmiPerMonth(double emiPerMonth) {
        this.emiPerMonth = emiPerMonth;
    }

    public LocalDate getLoanDisbursalDate() {
        return loanDisbursalDate;
    }

    public void setLoanDisbursalDate(LocalDate loanDisbursalDate) {
        this.loanDisbursalDate = loanDisbursalDate;
    }

    public int getRepaymentFrequecy() {
        return repaymentFrequecy;
    }

    public void setRepaymentFrequecy(int repaymentFrequecy) {
        this.repaymentFrequecy = repaymentFrequecy;
```

```
    }

    public  void calculateEMI(double rate, int installmentsPerYear, int
totalInstallments, double loanAmount, double residualValue) {
        double roiPerInst = rate / (installmentsPerYear * 100);
        double powerVal = Math.pow((1 + (roiPerInst)), totalInstallments);
        double emi= ((loanAmount * (roiPerInst)) - ((residualValue *
(roiPerInst)) / powerVal)) / (1 - (1 / powerVal));
        System.out.println(emi);
    }

    public  void generateRepaymentSchedule(double openBalance, double rate,
double numberOfInstallment, double tenure, double installmentAmount, double
inteterComponent){
        for(int i=1;i<numberOfInstallment;i++){
            double openingAmount = openBalance-((numberOfInstallment-1) *
installmentAmount);
            double intComponent = openBalance * (rate / 100) * (1.0 / 12);
            double principalCompo = installmentAmount - inteterComponent;
            if(openingAmount < 0){
                break;
            }

            System.out.println(i + "   |   " + openingAmount + "   |   " +
intComponent + "   |   " + principalCompo + "   |   " + installmentAmount +
"\n");
        }
    }

    public  double LatePenalty(LocalDate currentDate){
        if(currentDate.compareTo(loanDisbursalDate)>0) {
            System.out.println("No Late Penalty");
        }
        else
            System.out.println("Late Penalty will be charged");
        return 0;
    }

    public  double loanToValueRatio(double loanAsked, double propertyValue)
{
        return (loanAsked / propertyValue)*100;
    }

    @Override
    public String toString() {
        return "LoanAgreement{" +

                ", loanDisbursalDate=" + loanDisbursalDate +
                ", income=" + income +
                ", loanAgreementId=" + loanAgreementId +
                ", loanAmount=" + loanAmount +
                ", tenure=" + tenure +
                ", roi=" + roi +
                ", emiPerMonth=" + emiPerMonth +
                ", repaymentFrequecy=" + repaymentFrequecy +
                '}';
    }
}
```

Customer:

```
package org.example;

import java.time.LocalDate;

public class Customer {

    private int customerId = 0;
    private String customerName;
    private LocalDate dateOfBirth;
    private String emailAddress;

    public Customer(String test1, int i, int i1) {
    }

    public int getCustomerId() {
        return customerId;
    }

    public void setCustomerId(int customerId) {
        this.customerId = customerId;
    }

    private double monthlyIncome;
    private String profession;
    private double totalMonthlyExpenses;
    private double maxEligibleLoanAmount;
    private String designation;
    private String companyName;
    static int count = 0;

    Customer() {

        count++;
    }

    public double getMonthlyIncome() {
        return monthlyIncome;
    }

    @Override
    public String toString() {
        return "Customer{" +
                "customerId=" + customerId +
                ", customerName='" + customerName + '\'' +
                ", dateOfBirth=" + dateOfBirth +
                ", emailAddress='" + emailAddress + '\'' +
                ", monthlyIncome=" + monthlyIncome +
                ", profession='" + profession + '\'' +
                ", totalMonthlyExpenses=" + totalMonthlyExpenses +
                ", maxEligibleLoanAmount=" + maxEligibleLoanAmount +
                ", designation='" + designation + '\'' +
                ", companyName='" + companyName + '\'' +
                '}';
    }

    Customer(int customerId, String customerName,
             LocalDate dateOfBirth, String emailAddress, double
                     monthlyIncome, String profession, double
                     totalMonthlyExpenses, String designation, String
```

```
                    companyName) {
        this.customerId = customerId;
        this.customerName = customerName;
        this.dateOfBirth = dateOfBirth;
        this.emailAddress = emailAddress;
        this.monthlyIncome = monthlyIncome;
        this.profession = profession;
        this.totalMonthlyExpenses = totalMonthlyExpenses;
        this.designation = designation;
        this.companyName = companyName;
        count++;
    }

    public static void display() {
        System.out.println("The number of objects created of customer are:
" + count);
    }

}
```

RepaymentSchedule:

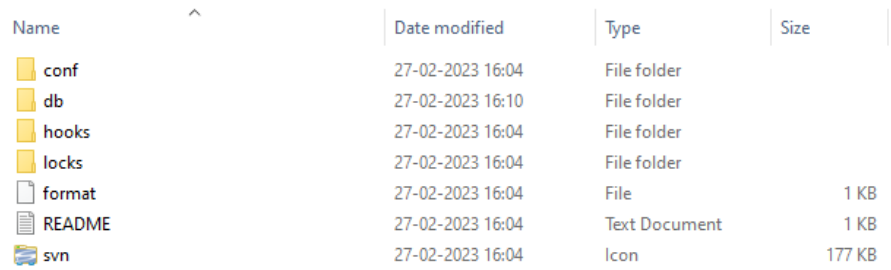
```
package org.example;

public class RepaymentSchedule {
    private double in;
    private double pn;

    private double outstandingPrincipal;

    public double getIn() {
        return in;
    }

    public void setIn(double in) {
        this.in = in;
    }

    public double getPn() {
        return pn;
    }

    public void setPn(double pn) {
        this.pn = pn;
    }

    public double getOutstandingPrincipal() {
        return outstandingPrincipal;
    }

    public void setOutstandingPrincipal(double outstandingPrincipal) {
        this.outstandingPrincipal = outstandingPrincipal;
    }
```

```
    public void setIn(double opn, double rate) {
        this.in = (opn * rate)/1200;
    }
}
```

OUTPUT:

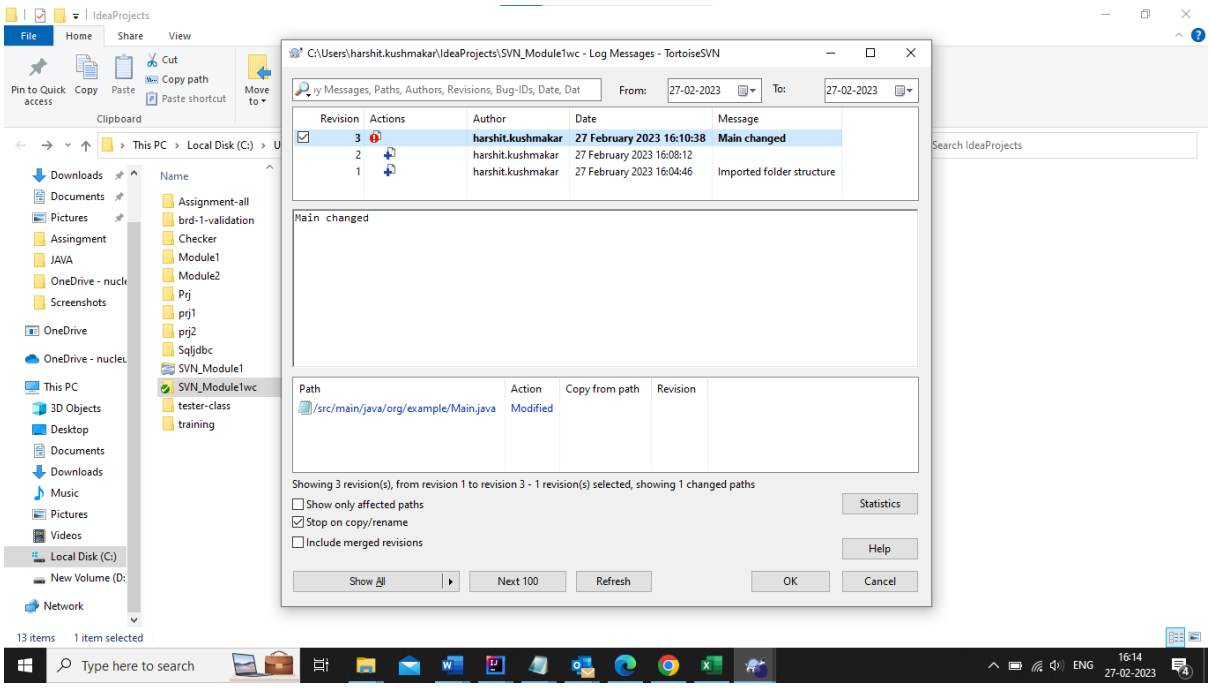
```
"C:\Program Files\Java\jdk-11\bin\java.exe" "-jav
false

Process finished with exit code 0
```

3. Import this Project using SVN and perform the checkout, update, revert commands on it.



Name	Date modified	Type	Size
conf	27-02-2023 16:04	File folder	
db	27-02-2023 16:10	File folder	
hooks	27-02-2023 16:04	File folder	
locks	27-02-2023 16:04	File folder	
format	27-02-2023 16:04	File	1 KB
README	27-02-2023 16:04	Text Document	1 KB
svn	27-02-2023 16:04	Icon	177 KB

Update



Revert