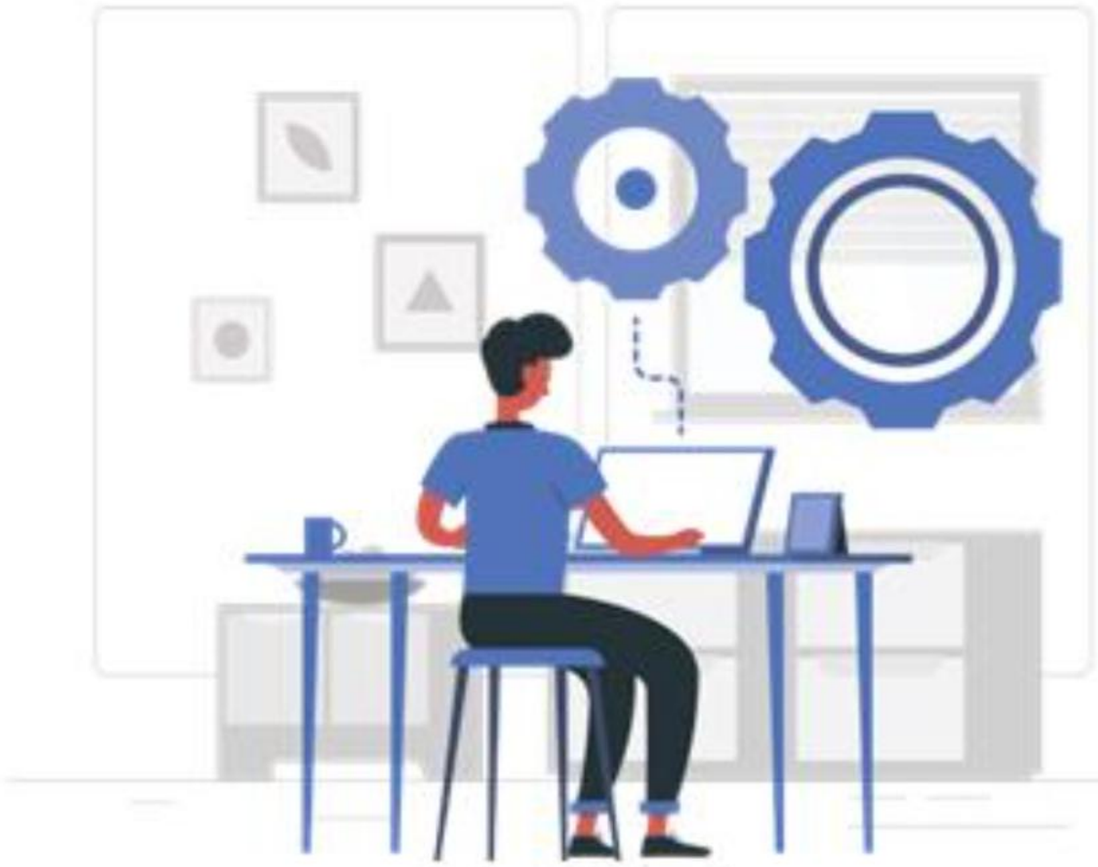


Practice
Implement Exception Handling

Practices

- Practice 1: Divide Number Game
- Practice 2: Company Survey

Implementation Environment

- The practices must be done in the IntelliJ IDE.
- Click here to install IntelliJ
- You must have access to GitLab.
- Install git to be able to clone and push code to the repository.
- You must be familiar with forking and cloning a git repository.

Implementation Environment

- Cloning a repository from git
  - After forking the boilerplate into your namespace, execute the below command on git bash or command prompt to clone the repository into your local machine.

```
git clone <repository link>
```

- Push the solution back to git
  - After completion of the practice, push the code back to git using the below commands.

```
git add .
git commit –m “comments on the push”
git push –u origin master
```

PRACTICE

Practice 1: Divide Number Game

Ron and Johnny are playing the divide number game, where one person gives two numbers, and the other person must divide both the numbers and give the remainder as the output.

Help Ron achieve this task.

Instructions for the Practice

- Click on the boilerplate.
- Fork the boilerplate using the fork button

- Select your namespace to fork the project.
- Clone the project into your local system.
- Open the project in the IntelliJ IDE.
- Work on the solution.
- Execute the test cases given in the test folder.
- Push the solution to git.

Tasks

- Write all the logic for the program inside the `GameDemo` class provided in the boilerplate.

- Write the logic to divide two numbers and return the remainder of the numbers inside the below method:

```
public String divideByNumber(int firstnumber,int secondNumber)
```

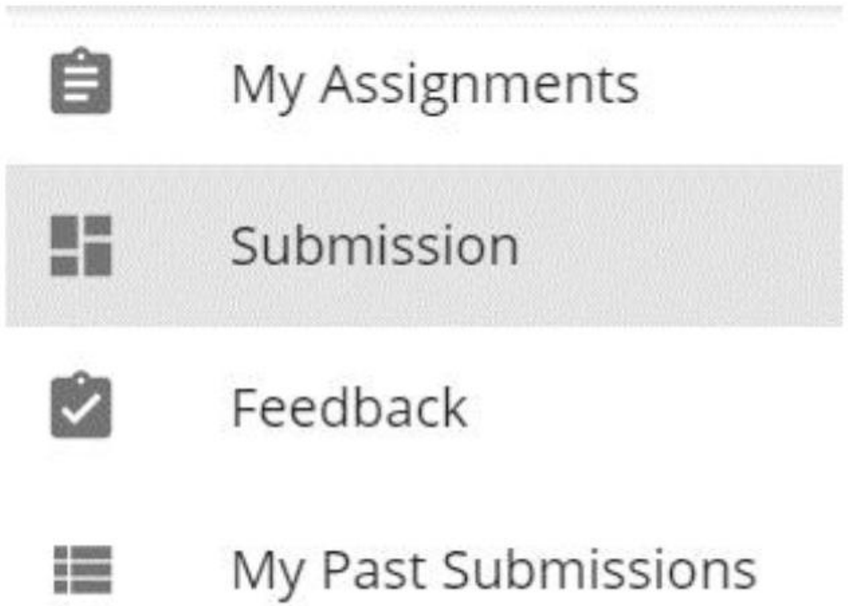
- Write try andcatch blocks to handle the exceptions.

Input and Output

- **Sample Input – 1:**
  - `int firstNumber = 10, int secondNumber = 5`
- **Sample Output – 1:**
  - `String remainder = 2`
- **Sample Input – 2:**
  - `Int firstNumber = 10 , int secondNumber = 0`
- **Sample Output – 2:**
  - `java.lang.ArithmeticException: /by zero`

Submission Instructions

- Submit the practice on Hobbes.
- Login to hobbes using your credentials.
- Click on **Submission** in the left navigation bar.



Submission Instructions (Cont'd)

- The **Submit for evaluation** page is opened.
- Under **Assignment Repository**, select the solution repository against which your submission will be evaluated.

Submission Instructions (Cont'd)

- Select your solution repository under **Search Submission Repo**

- Click on **Submit**.
- The results can be viewed in the **Past Submissions** screen.

PRACTICE

Practice 2: Company Survey

A company wants to find out the average age of its employees. The age data shared with the HR team is in raw format (age can be given in numbers or as a String). Help the team determine the average age of the employees.

Instructions for the Practice

- Click here for the boilerplate.
- Fork the boilerplate using the fork button

- Select your namespace to fork the project.
- Clone the project into your local system.
- Open the project in the IntelliJ IDE.
- Work on the solution.
- Execute the test cases given in the test folder.
- Push the solution to git.

Tasks

- Write all the logic for the program inside the `AverageAge` class provided.

- Write the logic to calculate the average age of the employees inside the method below and return the String:

```
public String averageAgeCalculator(String[] ageRawData)
```

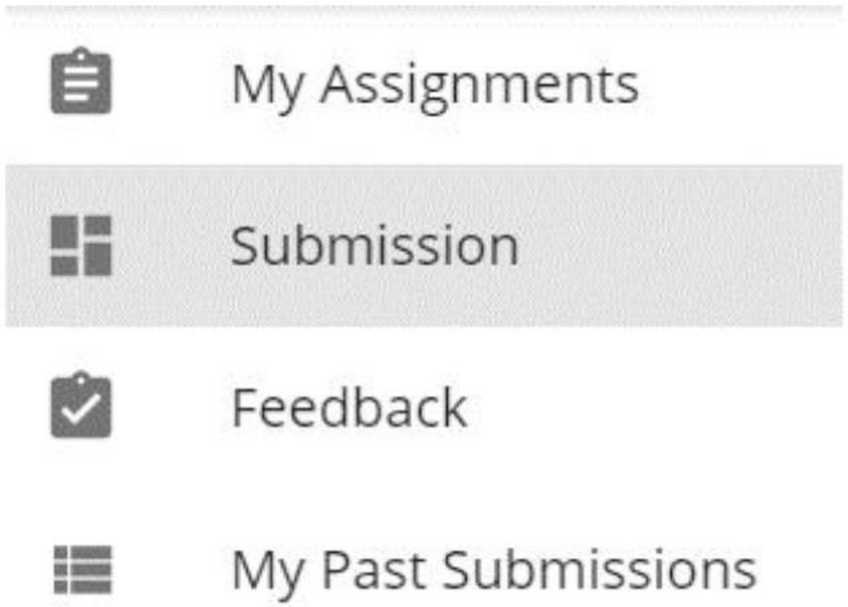
- Write try and catch to handle the appropriate exception.

Input and Output

- **Sample Input – 1:**
  - `String age[] = {"35","26","32","44"};`
- **Sample Output – 1:**
  - `Average Age - 34.25`
- **Sample Input – 2:**
  - `String age[] = {"35","26","thirty-two","44"};`
- **Sample Output – 2:**
  - `java.lang.NumberFormatException: For input string: "thirty-two"`

Submission Instructions

- Submit the practice on Hobbes.
- Login to hobbes using your credentials.
- Click on **Submission** in the left navigation bar.

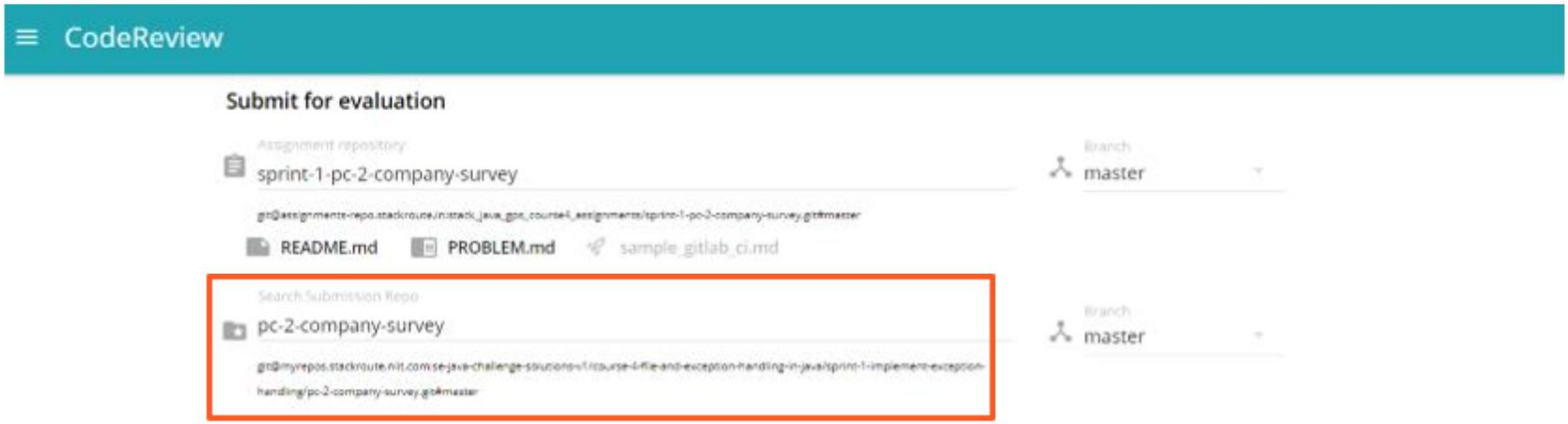


Submission Instructions (Cont'd)

- The **Submit for evaluation** page is opened.
- Under **Assignment Repository**, select the solution repository against which your submission will be evaluated.

Submission Instructions (Cont'd)

- Select your solution repository under **Search Submission Repo.**



- Click on **Submit**.
- The results can be viewed on the **Past Submissions** screen.