PRANEETH JADCHERLA

STUDENT ID: 209052391

FSIS ASSIGNMENT 2 REPORT

Loan Repayment Schedule Calculator

Introduction:

This is a small web application of Loan Calculator which displays a calculator where user provides few inputs and the loan repayment schedule is calculated on a monthly basis and displayed on the UI.

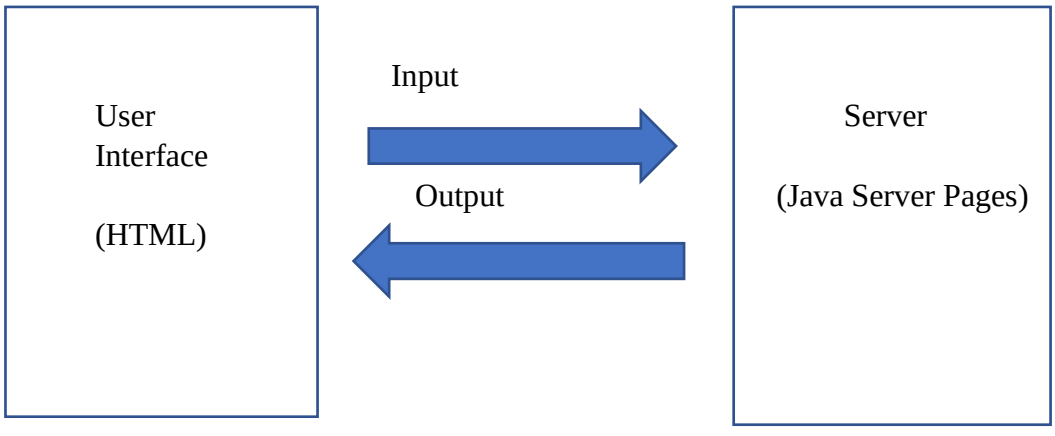
Functionality:

- User enters the loan Amount, Annual interest rate, loan period in months on the user interface.
- On clicking the submit button, The UI displays the loan amortization schedule in a table format which consists of below attributes:
    1. Payment No
    2. Payment Amount
    3. Principal Amount Paid
    4. Interest Amount Paid
    5. Loan Outstanding Balance

Technologies Used:

- HTML
- Java Script
- Java, JSP
- IDE – Eclipse
- Tomcat server 10.0

Architecture:

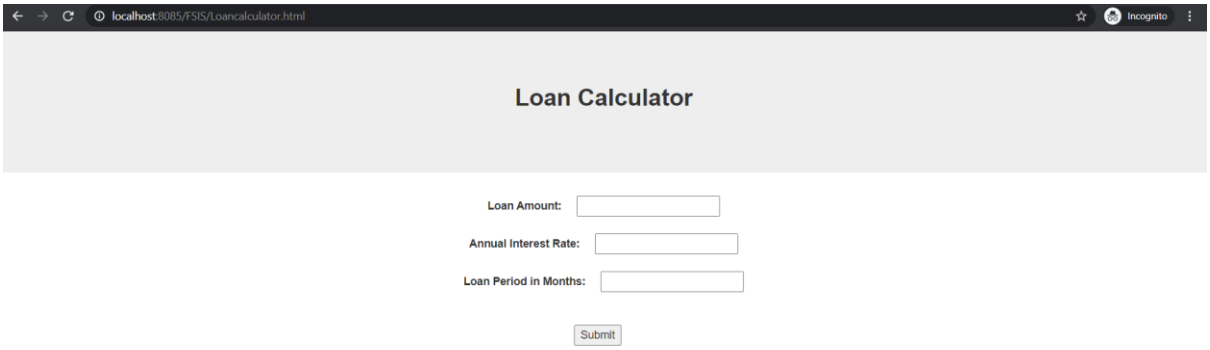


Implementation:

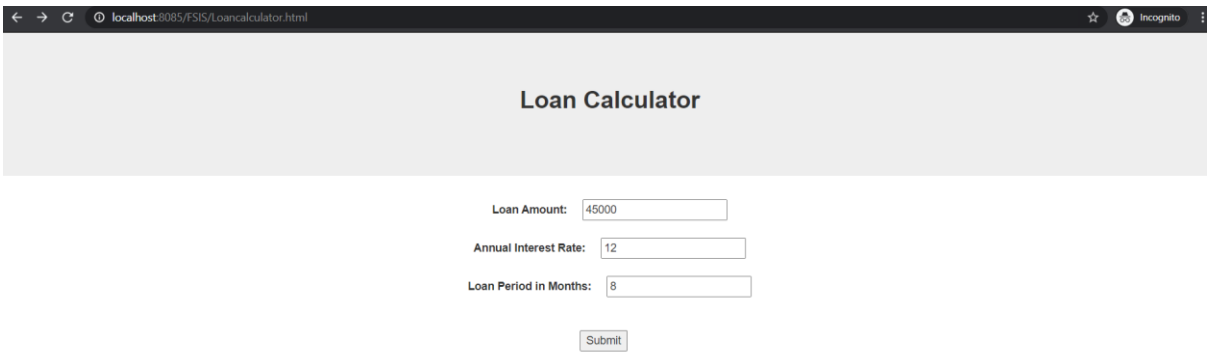
Functionality:

This sections gives an overview of the application functionality with screenshots.

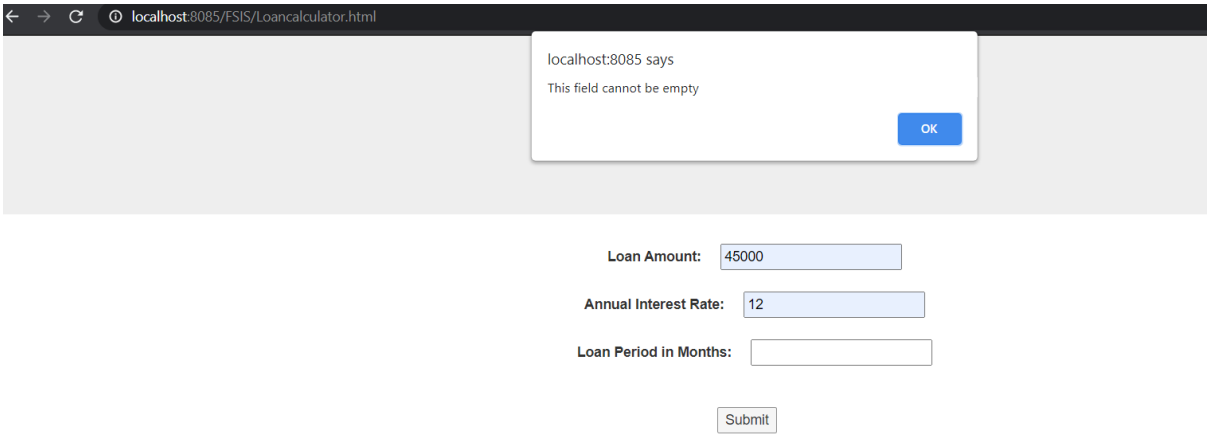
1. Home Page displaying Loan calculator

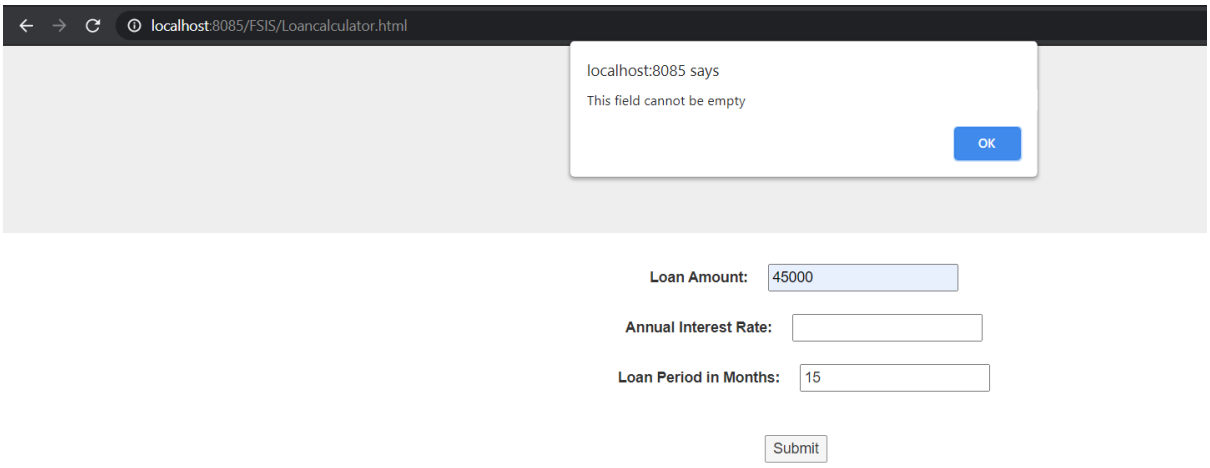


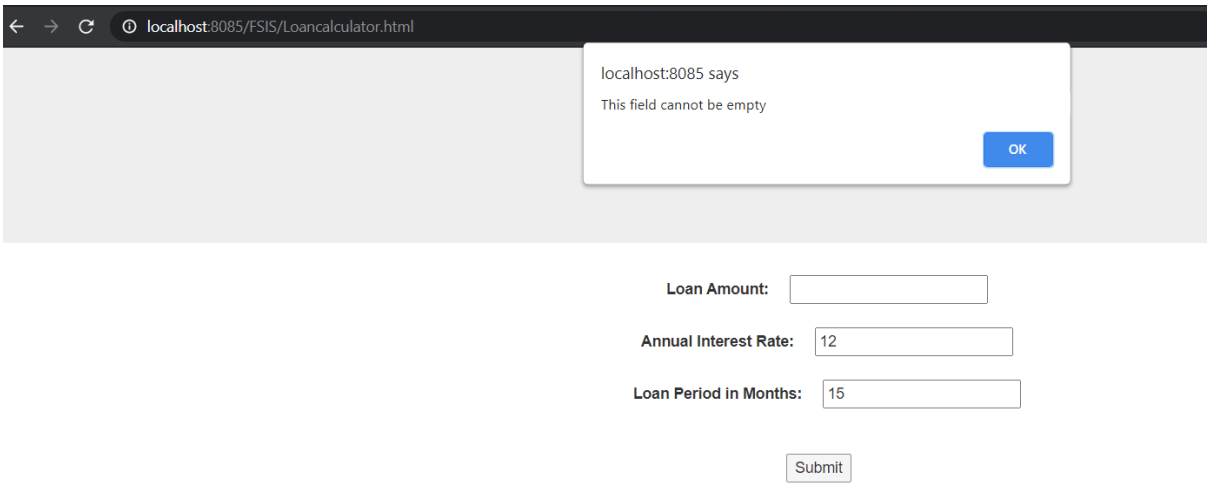
2. User entering required fields



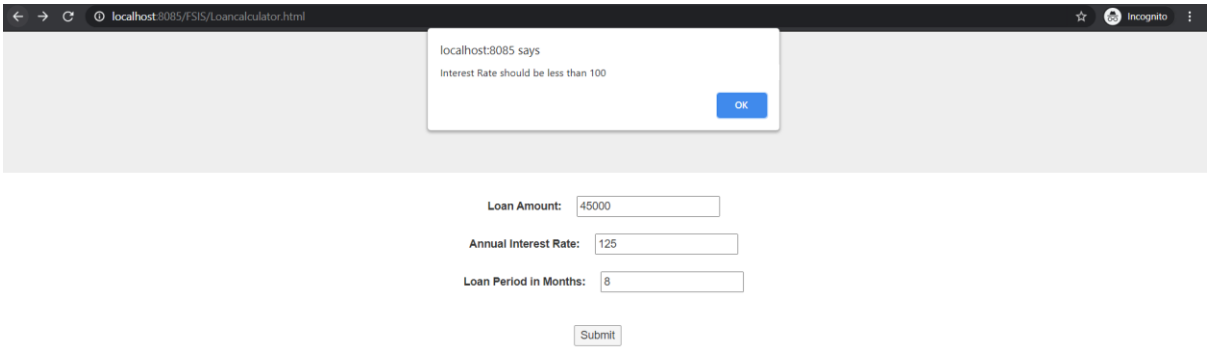
3. Validation checking if user is entering empty values



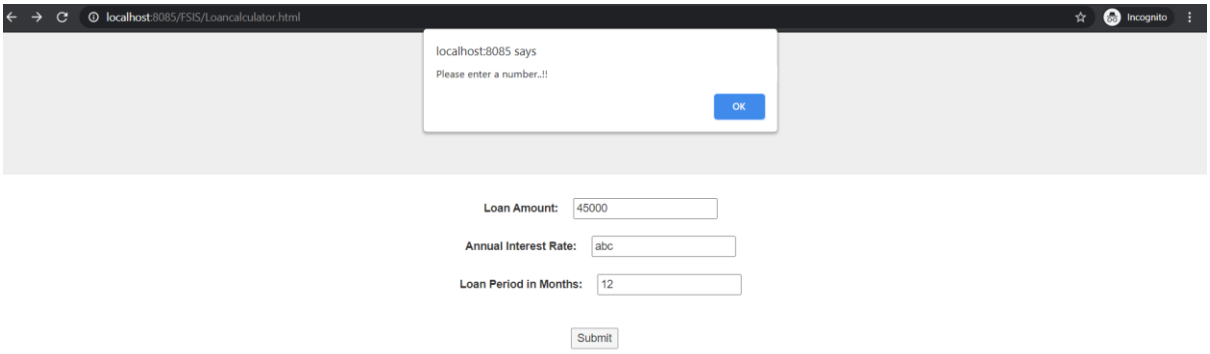


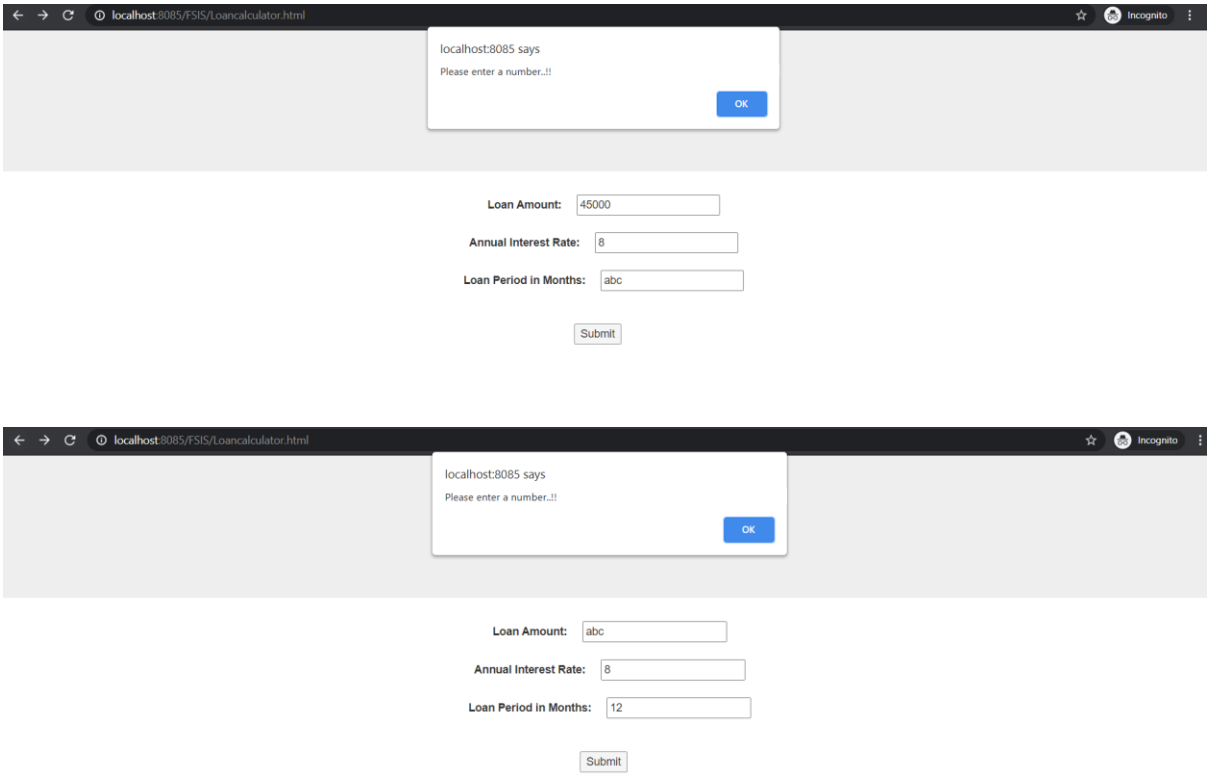


4. Validation checking if user is entering interestrate greater than 100

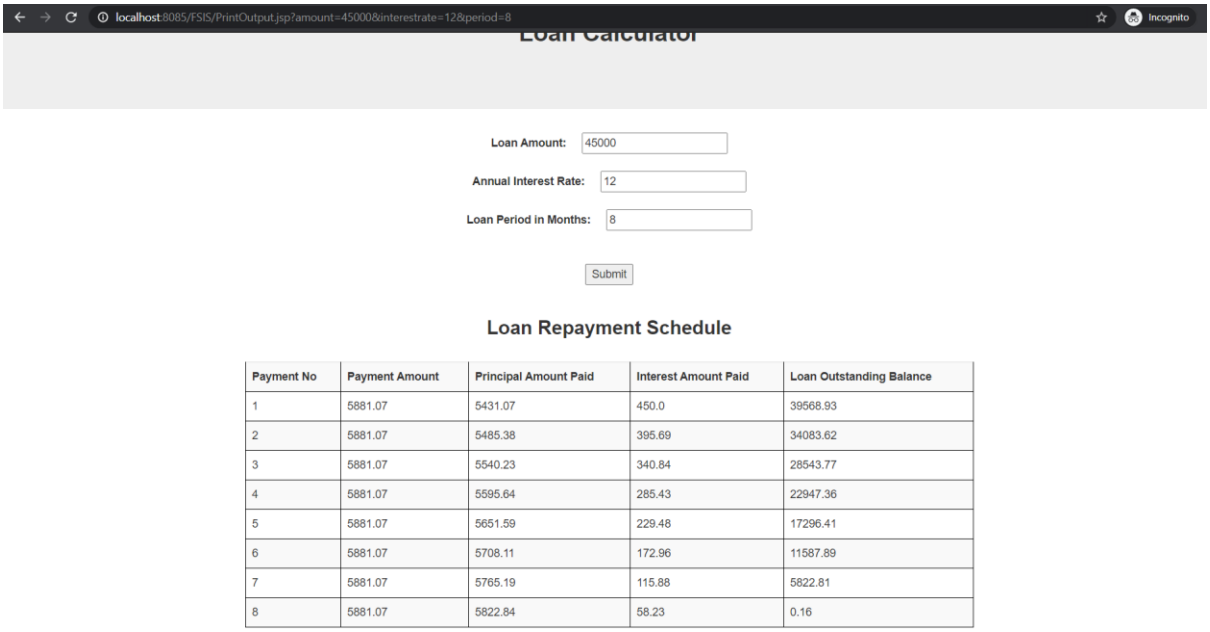


5. Validation checking if user is entering a number or not





6. Loan repayment schedule displayed after clicking submit button



Payment No	Payment Amount	Principal Amount Paid	Interest Amount Paid	Loan Outstanding Balance
1	5881.07	5431.07	450.0	39568.93
2	5881.07	5485.38	395.69	34083.62
3	5881.07	5540.23	340.84	28543.77
4	5881.07	5595.64	285.43	22947.36
5	5881.07	5651.59	229.48	17296.41
6	5881.07	5708.11	172.96	11587.89
7	5881.07	5765.19	115.88	5822.81
8	5881.07	5822.84	58.23	0.16

Coding:

This section gives a picture of main parts of the code and the logic implemented along with the code snippets.

Loancalculator.html

This file is the start of the application and has user interface where user provides all the inputs for Loan Amount, interest rate, loan period in months.

Loancalculator.html

```
<!DOCTYPE html>
<html>
<head>
<title>Loan Calculator</title>
<meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
</head>

<body>
<div class="jumbotron text-center">
<h2><b>Loan Calculator</b></h2><br>
</div>

<form action="PrintOutput.jsp" class="text-center" id="inputform" onsubmit="return validateForm()">
    <label for="amount">Loan Amount:</label>    
    <input type="text" id="amount" name="amount"><br><br>
    <label for="interest">Annual Interest Rate:</label>    
    <input type="text" id="interest" name="interestrate"><br><br>
    <label for="period">Loan Period in Months:</label>    
    <input type="text" id="period" name="period"><br><br>
    <br>
    <button>Submit</button>    
  </form>

```

PrintOutput.jsp

This is the JSP file which takes inputs from html file, processes the inputs and gives back the output in a tabular format.

PrintOutput.jsp

```
    <div class="container">
        <table style="width: 80%" class="table table-striped center">
            <tr>
                <th>Payment No</th>
                <th>Payment Amount</th>
                <th>Principal Amount Paid</th>
                <th>Interest Amount Paid</th>
                <th>Loan Outstanding Balance</th>
            </tr>
            <%
            Loancalculation lc = new Loancalculation();
            for (int n = 1; n <= period; n++) {
                float Payment = lc.PaymentAmount(amount, interestrate, period);
                float PP = lc.PrincipleAmountPaid(interestrate, period, n);
                float Interest = lc.InterestAmountPaid();
                float Balance = lc.LoanOutstandingBalance(interestrate);
            %>

            <tr>
                <td>
                    <%
                    out.println(n);
                    %>
                </td>
                <td>
                    <%
                    out.println(Payment);
                    %>
                </td>
                <td>
                    <%
                    out.println(PP);
                    %>
                </td>
                <td>
                    <%
                    out.println(Interest);
                    %>
                </td>
                <td>
                    <%out.println(Balance);%>
                </td>
```

Loancalculation.java

This is the java file which has the actual logic for calculating each value of the output such as Payment Amount , Principal Amount Paid, Interest Amount Paid and Loan Outstanding Balance.

Loancalculation.java

```
16    public float PaymentAmount(float A,float interestrate,float N) {
17
18        DecimalFormat df= new DecimalFormat("#.000");
19        DecimalFormat df2= new DecimalFormat("#.00");
20
21        R=interestrate/12/100;
22        R=Float.parseFloat(df.format(R));
23
24        P=(float)Math.pow((1+R),N);
25
26        Payment=(R*A*P)/(P-1);
27        Payment=Float.parseFloat(df2.format(Payment));
28        return Payment;
29
30    }
31
32    public float PrincipleAmountPaid(float interestrate,float N,int n) {
33
34        DecimalFormat df= new DecimalFormat("#.000");
35        DecimalFormat df2= new DecimalFormat("#.00");
36
37        R=interestrate/12/100;
38        R=Float.parseFloat(df.format(R));
39
40        P1=(float)Math.pow(1+R,1+N-n);
41        PP=(float)Payment/P1;
42
43        PP=Float.parseFloat(df2.format(PP));
44        return PP;
45    }
46
47    public float InterestAmountPaid() {
48
49        DecimalFormat df2= new DecimalFormat("#.00");
50        Interest=(float)Payment-PP;
51        Interest=Float.parseFloat(df2.format(Interest));
52        return Interest;
53    }
54
55    public float LoanOutstandingBalance(float interestrate) {
56
57        DecimalFormat df= new DecimalFormat("#.000");
58        DecimalFormat df2= new DecimalFormat("#.00");
```

Description of the Methods used in Java Program:

PaymentAmount()

This method calculates the Payment amount that needs to be paid by the user every month. I have taken the below formula to calculate:

`Payment=(R*A*P)/(P-1);`

`Where R is the rate of interest per month`

`A is the loan amount entered by the user`

`P= exp(`(1 + R), N) , where N is the loan period in months entered by user

PrincipalAmountPaid()

This method calculates the principal amount paid which is included in the payment amount. This will be calculated per amount and the formula is below

```
P1=(float)Math.pow(1+R,1+N-n);
PP=(float)Payment/P1;
```

Where n denotes each month starting from 1st month and PP is the principal amount paid

InterestAmountPaid()

This method is used to calculate the interest amount paid in the payment amount every month and the formula is below

```
Interest=(float)Payment-PP;
```

LoanOutstandingBalance()

This method is used to calculate the balance loan amount need to be paid after paying each month payment amount and the formula is below

```
Balance=(Interest/R)-PP;
```

Steps to run the application:

I have created a WAR file out of my application and below are the steps to be followed to run the WAR file

$CATALINA_HOME is the home directory where Apache Tomcat is installed. In my machine it is at below location.

C:\Program Files\Apache Software Foundation\Tomcat 10.0

1. Install Tomcat Server on your machine from below link

https://tomcat.apache.org/download-10.cgi

2. Once tomcat is downloaded navigate to below directory and place the WAR file of my application in below directory.
$CATALINA_HOME/webapps

2. Start the server by navigating to below folder and clicking on Tomcat application icon
$CATALINA_HOME/bin

3. Once you start the server you should be able to run the WAR file by navigating to below path.

The tomcat server usually runs on either 8080 or 8005 or 8085 port. When you start the server we can see the port on which tomcat is running.

http://localhost:8085/FSIS/Loancalculator.html