

# **PUNE INSTITUTE OF COMPUTER TECHNOLOGY**



**Department of Computer Engineering**  
(2021- 2022)

**LP-II**

**Batch: - K1**

**Calculator on Salesforce Cloud**

31107 - Sushant Sudhakar Awathare

**Guided by: - Prof. Madhuri Mane**

## **Table of Contents: -**

<b>Sr.no</b>	<b>Title</b>	<b>Page no.</b>
<b>1</b>	<b>Problem Statement</b>	<b>3</b>
<b>2</b>	<b>Motivation</b>	<b>3</b>
<b>3</b>	<b>Scope</b>	<b>3</b>
<b>4</b>	<b>Objective</b>	<b>3</b>
<b>5</b>	<b>Outcomes</b>	<b>3</b>
<b>6</b>	<b>Software and Hardware Requirements</b>	<b>3</b>
<b>7</b>	<b>Tech Stack</b>	<b>4</b>
<b>8</b>	<b>Screenshots of UI</b>	<b>5</b>
<b>9</b>	<b>Conclusion</b>	<b>6</b>

## **1. Problem Statement:**

Design and develop custom Application (in this case) Calculator using Salesforce Cloud.

## **2. Motivation**

- 1) Nowadays, the world has become online in many ways. In short, we can say the digital world. So, the Calculator application on the salesforce cloud will help users/customers to calculate complex operations in less time.
- 2) To learn development in cloud computing.

## **3. Scope**

Everyone likes that their work must be completed in less amount of time and the calculator on the salesforce cloud will help users/customers to do that. Hence, many users will use such applications which help them to achieve their goals in less time.

## **4. Objective:**

- To create the calculator on the salesforce cloud.
- To learn the basics and implement Apex programming language on Salesforce platform

## **5. Outcomes:**

- To get the output of operations which the user wants to perform.
- A new experience on developing the calculator on the Salesforce cloud and learning and understanding cloud computing.

## **6. Software and Hardware Requirements:**

Software:

- Windows 10 OS, 64 bits
- Salesforce cloud website.

Hardware:

- Processor: Intel i-5 8<sup>th</sup> gen
- Manufacturer: Acer Aspire-5
- Ram: 8 GB/ 16GB Optane memory

## 7. Tech Stack

- Salesforce cloud platform
- Apex programming language
- Java

#code:

Sample.apxc

```
public class Sample
{
    public Double val1 {get;set;}
    public Double val2 {get;set;}
    public Double result {get;set;}
    public String func {get;set;}

    public Sample()
    {
    }

    public void find()
    {
        if(func == 'add')
        {
            result = val1 + val2;
```

```

    }

    else if(func == 'sub')

    {

        result = val1 - val2;

    }

    else if(func == 'div')

    {

        result = val1 / val2;Apex:

    }

    else if(func == 'mul'){

        result = val1 * val2;

    }

    else

    {

        Integer temp = math.mod(Integer.valueOf(val1) , Integer.valueOf(val2));

        result = Double.valueOf(temp);

    }

}

}

```

Visual.vfp

```

<apex:page controller="Sample">

<apex:form >

<apex:pageBlock >
    <apex:pageBlockSection >
        <apex:pageBlockSectionItem >
            <apex:outputLabel value="Principal Amount"/>
        </apex:pageBlockSectionItem>
        <apex:pageBlockSectionItem >
            <apex:inputText value="{!p}"/>
        </apex:pageBlockSectionItem>
        <apex:pageBlockSectionItem >
            <apex:outputLabel value="No of Years"/>
        </apex:pageBlockSectionItem>
    </apex:pageBlockSection >
</apex:pageBlock >

</apex:form >

</apex:page>

```

```

    <apex:pageBlockSectionItem >
        <apex:inputText value="{!n}"/>
    </apex:pageBlockSectionItem>
    <apex:pageBlockSectionItem >
        <apex:outputLabel value="Rate of Interest"/>
    </apex:pageBlockSectionItem>
    <apex:pageBlockSectionItem >
        <apex:inputText value="{!r}"/>
    </apex:pageBlockSectionItem>
    <apex:pageBlockSectionItem >
        <apex:outputLabel value="Simple Interest "/>
    </apex:pageBlockSectionItem>
    <apex:pageBlockSectionItem >
        <apex:inputText value="{!result}" id="res"/><apex:actionStatus id="sts" startText="Calculating..."/>
    </apex:pageBlockSectionItem>
</apex:pageBlockSection>
<apex:pageBlockButtons >
    <apex:commandButton value="Calculate Simple Interest" action="{!find}" reRender="res" status="sts"/>
</apex:pageBlockButtons>
</apex:pageBlock>

</apex:form>
</apex:page>

```

# Screenshots of UI:

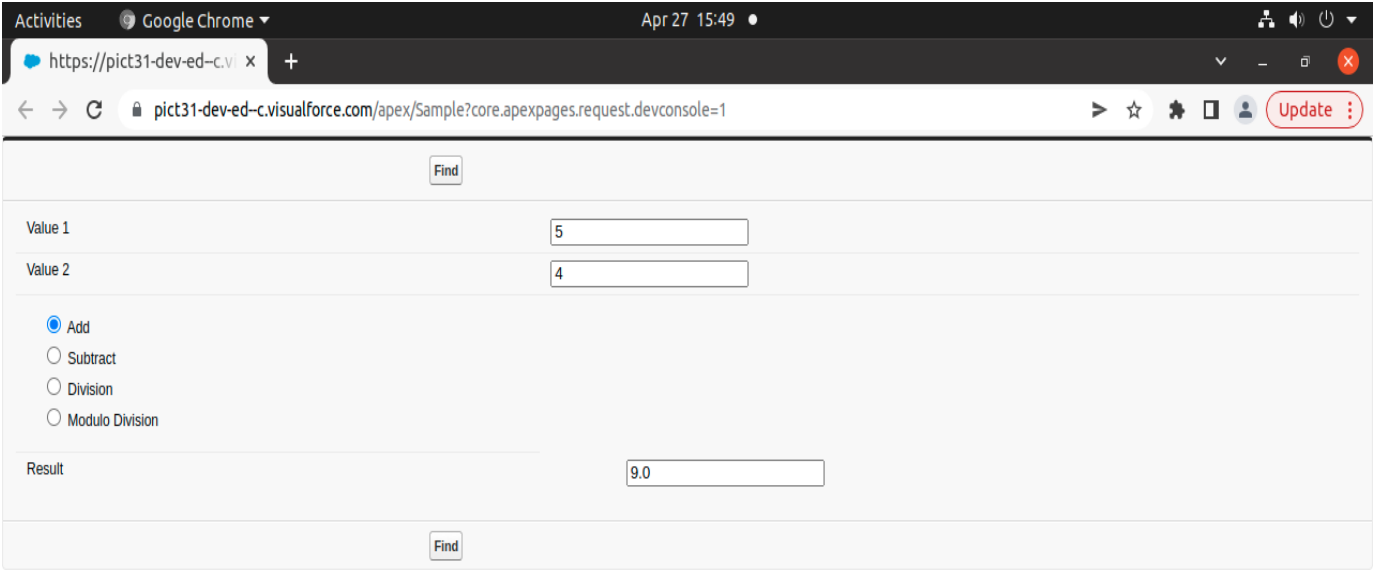


Fig:01

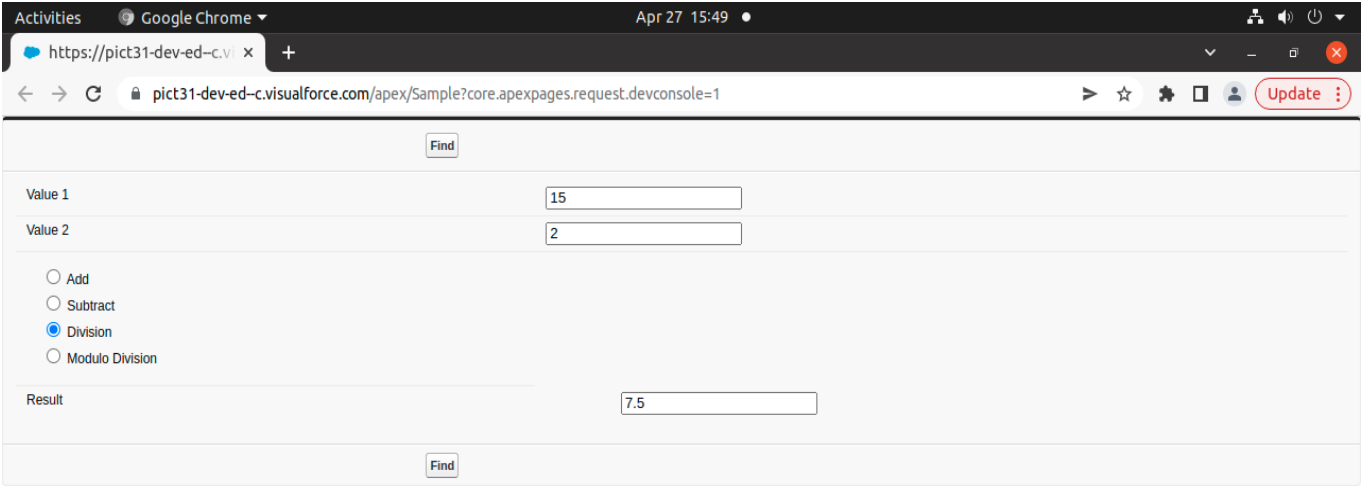


Fig: 02

## **7. Conclusion:**

Successfully completed a calculator on the Salesforce cloud platform using Apex programming language.