

# Software Re-Engineering      Final Exam

## (SE4001)

Date: May 25, 2024

Course Instructor(s)

Dr. Farooq Ahmed

Total Time (Hrs): 3

Total Marks: 60

Total Questions: 6

---

Roll No

---

Section

---

Student Signature

Do not write below this line

---

Attempt all the questions. In case of any ambiguity, take a valid assumption and state it clearly.

---

### **CLO 1: Describe Software Re-Engineering principles**

#### **Question 1**

**[5 marks]**

Which problem does Microservices architecture intend to solve in the context of Software Re-Engineering? Explain the role of standards in adopting a Microservices based architecture and how can it be beneficial?

---

### **CLO 2: Explain the activities involved in Software Re-Engineering**

#### **Question 2**

**[5+5 marks]**

Software Re-engineering is not just restricted to improvements in application design and architecture but also the data. Given that context, answer the following:

- (a) List the problems that can occur related to data re-engineering and which tool(s) can be useful for solving them?
- (b) List various types of NoSQL databases. Which type attempts to provide flexibility as well as consistency?

---

### **CLO 3: Apply different approaches to understand code**

#### **Question 3**

**[10 marks]**

CSS is a popular language to style HTML and other documents. It has a fairly simple syntax: each CSS rule comprises of a selector followed by a declaration block where all declarations are separated by a semicolon (;) and each declaration is a name-value pair, separated by colon (:). Semicolon is optional for last declaration. Some examples are explained as follows:

# National University of Computer and Emerging Sciences

## Lahore Campus

(i)	(ii)	(iii)
<pre>p {   color:red;   text-align:center; }</pre>	<pre>ul {   list-style-type:circle; }</pre>	<pre>img {   width:100px;   height:50px;   display:block }</pre>
<p><b>p</b> is an element selector that represents all paragraph (i.e. <code>&lt;p&gt;</code>) elements in document. So the rule states that all paragraphs shall be center aligned and red in color</p>	<p><b>ul</b> is an element selector that represents all unordered lists (i.e. <code>&lt;ul&gt;</code>) elements in document. So the rule states that all unordered lists shall use circle as a style symbol.</p>	<p><b>img</b> is an element selector that represents all images (i.e. <code>&lt;img&gt;</code>) elements in document. So the rule states that all images shall be displayed as a block of width 100 and height 50 pixels.</p>

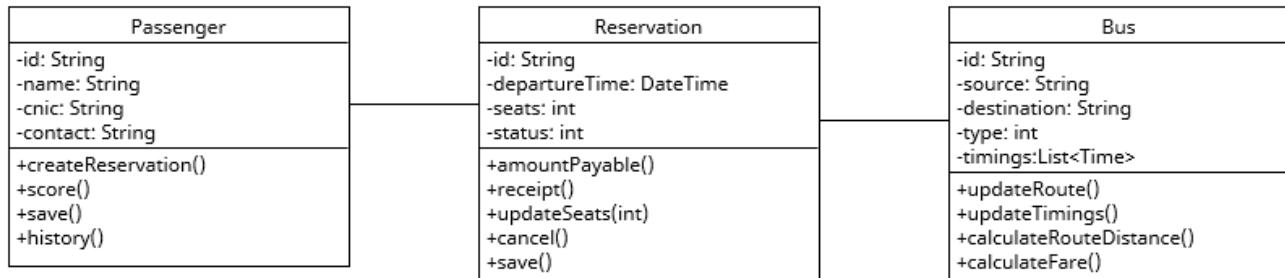
Assuming that there are no specific reserved words and all valid identifiers can be used as either selectors, names or values, **provide BNF grammar** for declaring a simple CSS rule. Note that number of declarations can be different.

### CLO 4: Compute metrics to analyze design documents

#### Question 4

[10 marks]

Consider the design for a bus reservation system below:



Bus class represents a transportation vehicle for a given route and timings. It is responsible for maintaining route and timings information, compute distance which helps compute fare amount. Type of bus (either standard or luxury) affects the calculation of fare. Moreover, calculation of distance is also a complex computation, accounting for different source and destination possibilities. It is assumed that the bus service operates in 3 cities only: Lahore, Islamabad and Peshawar.

Reservation class captures information for a reservation requested by a Passenger for a given bus on specified timings. Each reservation can have multiple seats. Reservation class is responsible for computing the amount payable by collaborating with relevant functionality (route and fare calculation) of Bus class.

Passenger class capture information of passenger(s) who intend to travel. It is responsible for creating a new reservation as well as maintaining historical record and a credit score (value of this passenger for company that is simply the total amount paid by customer, accumulating all trips).

Based upon the above description, compute the following metrics:

# National University of Computer and Emerging Sciences

## Lahore Campus

1. WMC for Bus class, while accounting for complexity of each method
2. RFC for Reservation class
3. RFC for Passenger class
4. CBO for Reservation class
5. CBO for Bus class

### CLO 5: Implement refactoring strategies for effective re-engineering

#### Question 5

[10+5 marks]

(a) Consider the following Java code:

<pre>class Employee {     private String name;     private Date joiningDate;     // ...      public Employee(String n,                     Date jd) {         this.name = n;         this.joiningDate = jd;     }      // ... }</pre>	<pre>class LeaveCalculator {      public int countAllowedLeaves(Employee e){         int casualLeaves = 0;         int earnedLeaves = 0;          Date today = new Date();         long diff = today.getTime() - e.getJoiningDate().getTime();         if ( diff &lt; 6*30*24*60*60*1000){             // 6 months probation phase             casualLeaves = 10;         }         else if (diff &gt; 6*30*24*60*60*1000 &amp;&amp;                 diff &lt; 365*24*60*60*1000){             // permanent with less than 1 year experience             casualLeaves = 15;         }         else{             // permanent with 1 year or more experience             casualLeaves = 15;             earnedLeaves = 21;         }          return casualLeaves + earnedLeaves;     }      public static void main(String args[]){         Employee e = new Employee("Asim",new Date(2023,10,1));         System.out.println(countAllowedLeaves(e));     } }</pre>
---	---

Show refactored code after applying the following refactorings:

- i. Decompose conditional
- ii. Replace Constructor with Factory

# National University of Computer and Emerging Sciences

Lahore Campus

(b) Which refactoring will break the following unit test?

```
@Test  
public void testAllowedLeaves(){  
    Employee e = new Employee("Asim", new Date(2023,10,1));  
    LeaveCalculator lc = new LeaveCalculator();  
    assertEquals(15, lc.countAllowedLeaves(e));  
}
```

Re-write the unit test code to fix the issue.

## Question 6

[10 marks]

Consider an XML based representation of a data model that needs to be converted into equivalent SQL DDL statements:

XML	SQL (DDL)
<pre>&lt;entities&gt;     &lt;entity name="Student" &gt;         &lt;attribute name="id" type="int" primarykey="true" /&gt;         &lt;attribute name="firstname" type="text" /&gt;         &lt;attribute name="lastname" type="text" /&gt;         &lt;attribute name="rollno" type="text" /&gt;         &lt;attribute name="batch" type="text" /&gt;     &lt;/entity&gt;      &lt;entity name="Course" &gt;         &lt;attribute name="id" type="int" primarykey="true" /&gt;         &lt;attribute name="title" type="text" /&gt;         &lt;attribute name="code" type="text" /&gt;         &lt;attribute name="type" type="text" /&gt;         &lt;attribute name="level" type="int" /&gt;     &lt;/entity&gt; &lt;/entities&gt;</pre>	<pre>CREATE TABLE Student (     id int PRIMARY KEY,     firstname text,     lastname text,     rollno text,     batch text )  CREATE TABLE Course (     id int PRIMARY KEY,     title text,     code text,     type text,     level int )</pre>

Write XSLT script for the transformation. Please note that in DDL last column / attribute is not followed by a comma (,). Also note that in XSLT there is an `<xsl:if>` element for decision making with the following syntax:

```
<xsl:if test="expression">  
    ...some output if the expression is true...  
</xsl:if>
```