SAVEETHA SCHOOL OF ENGINEERING
SIMATS, CHENNAI

Engineer to Excel

Course Code /Title: CSA4399 – Internet Programming
Programme : Computer Science and Engineering

ASSIGNMENT 4 QUESTIONS

S.No	Questions	Marks	CO	BTL
1	From a developer's perspective, discuss why JDBC is essential in building database-driven applications. How to achieve JDBC CONNECTION pooling using JDBC Data Source and JNDI in Apache Tomcat Server. Provide examples of executing SQL queries using JDBC statements. Discuss the differences between Statement, Prepared Statement, and Callable Statement.	20	CO5	3
2	Describe the lifecycle phases of a JSP page. Explain the significance of each phase in the JSP execution process. Discuss the different ways to embed Java code within a JSP page with examples. Explain the advantages and disadvantages of using scriptlets, declarations, and expressions in JSP.	20	CO4	2
3	You need to develop a PHP program that generates a chessboard using HTML tables. The table should have a total width of 400px, and each cell should have a height and width of 30px. The chessboard should alternate colors between black and white for each cell to represent a typical chessboard layout. How would you write a PHP program using nested for loops to create a chessboard? The chessboard should be displayed using an HTML table with a total width of 400px, and each cell should have a height and width of 30px. Explain how you would use the nested for loops to alternate the cell colors and ensure the chessboard pattern is correctly displayed. Provide the code for this program	20	CO4	2
	You are developing a PHP application that reads content from a text file and uses regular expressions to extract specific patterns, such as email addresses and phone numbers. After extracting the data, the application should store the results in a new XML file following a defined schema for the data. Additionally, you need to compare and contrast DTD (Document Type Definition) and XML Schema for defining the XML structure. How would you create a PHP application that reads content from a text file and uses regular expressions to extract specific patterns (e.g.,	20	CO5	2

SAVEETHA SCHOOL OF ENGINEERING
SIMATS, CHENNAI

Engineer to Excel

	email addresses and phone numbers)? After extracting the data, store the results in a new XML file following a defined schema for the data. Additionally, compare and contrast DTD and XML Schema for defining XML structure. Describe the steps and provide the code for the application.			

Assignment 04

Why JDBC is essential in building database driven applications.

JDBC is essential because it provides a standard API for java applications to interact with databases.

Achieving JDBC connection Pooling using JDBC

Configure Data Source

```
<Resource name="jdbc/myDB"
          auth="Container"
          type="javax.sql.DataSource"
          maxTotal="20"
          maxIdle="10"
          username="dbuser"
          password="dbPassword
          url="jdbc:mysql://localhost:3306/
               mydatabase"/>
```

Lookup Data Source in java code

```
import javax.naming.Context;
import javax.naming.InitialContext;
import javax.sql.DataSource;
import java.sql.Connection;
public class DatabaseUtil {
```

```
Context initContext = new InitialContext();
DataSource ds = (DataSource) initContext.
                lookup("java:/comp
return ds.getConnection();
}
}
```

Executing SQL queries using JDBC Statement

```
try (Connection con = DatabaseUtil.getConnection();
     Statement stmt = conn.createStatement();
     String query = "SELECT * FROM Users";
while (rs.next()){
}
}
```

O/P:-

Statement example O/P:-

User ID: 1, Name: Srinivas

User ID: 2, Name: Gunti

Prepared Statement

User ID: 1, Name: ~~Usha~~ Srinivas

Callable Statement

user ID: 1, Name: Srinivas

2) Life cycle phases of a JSP page

1) Translation phase
2) compilation phase
3) Initialization phase
4) Request processing phase
5) Destruction phase

Embedding Java code in JSP

① Scriptlets

<%int sum = 5+5; %>
<P> The sum is <%= sum %> </P>

o/p
The sum is : 15

② Declarations

<%! int add (int a, int b) { return a+b; } %>
<P> The result is: <%= add (3, 7) %> </P>

o/p:-
<P> current time: <%= new Java.util.Date() %> </P>

3) Expressions

<P> current time: <%= new Java.util.Date() %> </P>

o/p:-

current time: mon sep 09 09:30:00 PDT 20

Scriptlets:-

Advantages: Easy to use for embedding simple Java logic

Disadvantages:- Lead to messsy code, difficult to maintain

Declaration:

Advantage:- useful for declaring reusable methods and variables across multiple

Disadvantages: can clutter JSP with Java code, leading to poor separation of concerns.

3) Generates a chess board using html tables. width of 400px (total).

PHP code:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title> Chess Board </title>
    <style>
        table {
            width: 400px;
            border-collapse: collapse;
        }
    td {
```

```
        width : 30px;
        height : 30px;
    }
</style>
</head>
<body>
    <table>
        <?php
        // Loop for 8 rows
        echo "<tr>";
        for ($col = 0; $col < 8; $col++) {
            if (($row + $col) % 2 == 0) {
                echo "<td style='background-color: white;'></td>";
            } else {
                echo "<td style='background-color: black;'></td>";
            }
        }
        echo "</tr>";
    }
    ?>
    </table>
</body>
</html>
```

O/P:-

[ ] [#] [ ] [#] [ ] [#] [ ] [#]

[#] [ ] [#] [ ] [#] [ ] [#] [ ]

[ ] [#] [ ] [#] [ ] [#] [ ] [#]

[#] [ ] [#] [ ] [#] [ ] [#] [ ]

[ ] [#] [ ] [#] [ ] [#] [ ] [#]

[#] [ ] [#] [ ] [#] [ ] [#] [ ]

4) PHP Application to extract data and store in XML

Steps:-

i) Read content from a text File

ii) extract pattern using Regular

iii) create & store Results in an XML File

iv) Define XML schema.

PHP Code:-

```
<?php
$filename = "input.txt";
$content = file_get_contents($filename);
preg_match_all('/[a-zA-Z0-9_%+-]+@
[a-zA-Z0-9.-]+\.[a-zA-Z]
preg_match_all('/\+?[0-9]{1,3}[. \s]?[0-9]
{1,4}[-. \s]?[0-9]{3,4}/', $content,
```

```
        $email Element -> add child ("email", $email);
    }
    $phone Element = $xml -> add child (Phones);
    for each ($ phones [0] as $ phone ){
        $ phone element -> add child (Phone, $Phone)
    }
}
$xml -> as xml ('output.xml');
```

```
<data>
    <emails>
        <email> example @example.com </email>
        <email> example @example.com </email>
    </emails>
    <Phones>
        <Phone> +123-456-7890 </Phone>
        <Phone> 987-654-3210 </Phone>
    </Phone>
</data>
```