Assignment - 4

```
① <context>
      <Resource = name = jdbc/mydatasource",
   auth = "container"
   type = "javax.sql DataSource"
                          max Total = "20"
                          maxIdle = "10"
   maxwaitmillis = "10000".
   username = "dbuser".
   Password = "dbpassword".
   driverclassName = "com.mysql.jdbc.driver".
   url = "jdbc: mysql:// local host: 3306 / mydatabase
   />
   </context>
```

Collable Statement.

```
import java.sql collable statement;
import java.sql connection;
import java.sql: SQL Exception;
import java.sql. Types;
public class CollableStatement Example {.
Public void callstoredprocedure (int Employee id.
          String sql " { call get EmployeeName
                              (?, ?)}";
try(connection conn = Databaseutility. get
                    connection();
  employee Name = cstmt . getString(2);
  System.out.println( "Employee Name: "+ Employee Name);
   } catch(SQL Exception e){
} }  e.printStackTrace();
```

Output:-

Employee ID: 1, Name: John.

Employee ID: 2, Name: Jane.

Employee ID: 3, Name: Emily.

Rows updated: 1

Employee Name: John.

(2) Lifecycle of phase of a JSP page:-

1.) Translation phase:-

The JSP page is translated into a Java servlet by the JSP engine (e.g., HTML mixed with JSP tags).

Significance:-

This phase ensures that the JSP content is converted into a form that the Java servlet container can execute.

Compilation Phase:-

the java source code generated from the translation phase.

Compilation ensures that the JSP is converted to executable.

Initialization phase:-

The servlet container initializes the servlet in the servlet instance.

Request Processing Page:-

The servlet process incoming client requests by calling the service()
This phase is where the dynamic content generation occurs.

Destroy Phase:-

The servlet container, container destroy the servlet instance the destroy()
The phase ensure that resources are properly released.

(3) Php code:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport"
content="width=device-width, initial-scale;
<title>Chessboard</title>
<style>
    table {
        border-collapse:collapse;
        width:400px;
        height:400px;
    }
td {
    width:30px;
    height:30px;
for ($row=0; $row<8; $row++){
    echo "<tr>";
for ($col=0; $col<8; $col++) {
    echo "<td>";
```

```
}
?>
</frame>
</body>
</html>
```

Output :-

[w] [B] [w] [B] [w] [B] [w] [B]
[B] [w] [B] [w] [B] [w] [B] [w]
[w] [B] [w] [B] [w] [B] [w] [B].
[B] [w] [B] [w] [B] [w] [B] [w]
[w] [B] [w] [B] [w] [B] [w] [B].

* w represent a white cell.
* B represent a black cell

4) [illegible] for the Application

```
<?php
$textFilepath = "input.txt";
$xmlFilepath = "output.xml";
$textcontent = file_get_contents($textFilepath);
$emailpattern = "/[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}/";
$phonepattern = "/\b\d{10}\b/";
preg_match_all($emailpattern, $textcontent, $emails);
$simpleXML = new SimpleXMLElement("<Data/>");
$phoneElement = $xml->addChild [illegible]
echo "Data extracted and saved to XML file successfully.";
```