**PROTOCOLO INSTRUMENTO DE EVALUACIÓN**

|  |  |
| --- | --- |
| **SEDE:** | **Formosa** |
| **FACULTAD:** | **INGENIERÍA, TECNOLOGÍA Y ARQUITECTURA** |
| **CARRERA:** | **MODALIDAD**  ☐Presencial  ☐Digital |
| **NOMBRE DE LA ASIGNATURA: PROGRAMACIÓN III** | **MODALIDAD**  ☐Presencial  ☐Presencial Digital  ☐Presencial Mixta  ☐Digital |
| **NOMBRE Y APELLIDO DEL DOCENTE** | **Marcelo Alejandro Toledo** |
| **NOMBRE Y APELLIDO DEL ESTUDIANTE** | **Victoria Valentina Maidana Corti** |
| **DNI DEL ESTUDIANTE** | **43.938.790** |
| **FECHA Y HORARIO DE REALIZACIÓN Y/O DE ENTREGA: 24/05/2024 - 21 hs** | |

|  |  |  |
| --- | --- | --- |
| **TIPO DE EVALUACIÓN:** | ☐Trabajo Práctico I / 1er evaluación (Régimen especial)  ☐Trabajo Práctico II / 2da evaluación (Régimen especial)  ☐Evaluación Parcial / 3ra evaluación (Régimen especial)  ☐Evaluación Integradora / 4ta evaluación (Régimen especial) | |
| **MODALIDAD DE INSTANCIA EVALUATIVA** | ☐Individual  ☐Parejas  ☐Grupal | ☐Sincrónica  ☐Asincrónica |
| **CONTENIDOS A EVALUAR**  Eje Temático N° 1 a 3 | | |
| **CRITERIOS DE EVALUACIÓN**  1. Reconocer los elementos conceptuales.  2. Interpretar las situaciones problemáticas planteadas.  3. Identificar y representar las situaciones planteadas. | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CONSIGNAS DE EVALUACIÓN / ASIGNACIÓN DE PUNTAJE**   1. Nombrar tres Paradigmas de Programación, dar ejemplos. (1 punto)  |  |  | | --- | --- | | **Paradigma** | **Ejemplo** | | Imperativo | C, Python, Java | | Orientado a Objetos | Java, C++, Python | | Funcional | Haskell, Lisp, Erlang |  1. Explique que es la Programación concurrente, paralela y distribuida. (1 punto)  * **Programación Concurrente**: Permite ejecutar múltiples tareas al mismo tiempo dentro de un mismo programa. * **Programación Paralela**: Divide una tarea grande en subtareas más pequeñas que se ejecutan simultáneamente en múltiples procesadores. * **Programación Distribuida**: Utiliza múltiples sistemas independientes que trabajan juntos para resolver un problema.  1. **Planteamiento del problema**: un Laboratorio de Análisis Clínicos necesita implementar un nuevo módulo para su sistema informático. El objetivo de este es poder informatizar la planilla del Protocolo de Ingreso del Personal, actualmente se registra como muestra en la siguiente figura:      1. Crear:   Analizar y crear una base de datos y la/s tabla/s correspondientes para registrar el ingreso del personal. (1,5 puntos)   |  | | --- | | Parcial.sql  -- phpMyAdmin SQL Dump  -- version 5.2.1  -- https://www.phpmyadmin.net/  -- Servidor: 127.0.0.1  -- Tiempo de generación: 24-05-2024 a las 22:57:21  -- Versión del servidor: 10.4.32-MariaDB  -- Versión de PHP: 8.2.12  SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";  START TRANSACTION;  SET time\_zone = "+00:00";  /\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;  /\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;  /\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;  /\*!40101 SET NAMES utf8mb4 \*/;  --  -- Base de datos: `parcial`  -- Estructura de tabla para la tabla `personal`  CREATE TABLE `personal` (  `pers\_idpersonal` int(11) NOT NULL,  `pers\_nombre` varchar(65) DEFAULT NULL,  `pers\_apellido` varchar(65) DEFAULT NULL,  `pers\_fecha` date DEFAULT NULL,  `pers\_hora` time DEFAULT NULL,  `pers\_temperatura` tinyint(1) DEFAULT NULL,  `pers\_tos` tinyint(1) DEFAULT NULL,  `pers\_insuficienciaRespiratoria` tinyint(1) DEFAULT NULL,  `pers\_dolorDeGarganta` tinyint(1) DEFAULT NULL,  `pers\_perdidaDelOlfato` tinyint(1) DEFAULT NULL,  `pers\_perdidaDelGusto` tinyint(1) DEFAULT NULL,  `pers\_otrosSintomas` varchar(255) DEFAULT NULL,  `pers\_tuvoContactoAislamiento` tinyint(1) DEFAULT NULL,  `pers\_tuvoContactoUltDias` tinyint(1) DEFAULT NULL,  `pers\_lugarViaje` varchar(65) DEFAULT NULL,  `pers\_observaciones` varchar(255) DEFAULT NULL  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;  --  -- Índices para tablas volcadas  -- Indices de la tabla `personal`  --  ALTER TABLE `personal`  ADD PRIMARY KEY (`pers\_idpersonal`);  --  -- AUTO\_INCREMENT de las tablas volcadas  -- AUTO\_INCREMENT de la tabla `personal`  ALTER TABLE `personal`  MODIFY `pers\_idpersonal` int(11) NOT NULL AUTO\_INCREMENT;  COMMIT;  /\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;  /\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;  /\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/;  **Imagen de la table en MYSQL** |  1. Desarrollar:   Un módulo que permita dar de alta, baja y modificar (ABM), registros de la tabla creada.  Copiar en este documento las líneas de código donde incluyan la conexión a la base de datos, alta, baja y modificación (ABM).  (6,5 puntos)   |  |  | | --- | --- | | **Conexión** | <?php  function conectarBD()  {  $bd = mysqli\_connect('localhost', 'root', '', 'parcial');  if (!$bd) {  echo ('No se pudo conectar a la base de datos');  }  return $bd;  } | |  | | | **Crear** | <?php include 'conexion.php';  $bd = conectarBD();  if ($\_SERVER["REQUEST\_METHOD"] == "POST") {  $nombre = $\_POST['nombre'];  $sql = "INSERT INTO personal (pers\_nombre) VALUES ('$nombre')";  if ($bd->query($sql) === TRUE) {  header('Location: index.php');  } else {  echo "Error: " . $sql . " " . $bd->error;  }  }  ?>  <form method="post">  <label for="nombre">Nombre</label>  <input required name="nombre" class="form-input" type="text" id="nombre">  <button type="submit" value="Crear">Crear</button>  </form> | |  | | | **Eliminar** | <?php  function conectarBD()  if ($\_SERVER["REQUEST\_METHOD"] == "POST") {  $id $\_POST['id']  $sql = "DELETE FROM personal WHERE pers\_id = $id";  if ($bd->query($sql) === TRUE) {  header('Location: index.php');  } else {  echo "Error: " . $sql . " " . $bd->error;  }  }  ?>  <form method="post">  <label for="id">ID</label>  <input required name="id" class="form-input" type="text" id="id">  <button type="submit" value="Eliminar">Eliminar</button>  </form> | |  | | | **Modificar** | <?php include 'conexion.php';  $bd = conectarBD();  if ($\_SERVER["REQUEST\_METHOD"] == "POST") {  $id = $\_POST['id'];  $nombre = $\_POST['nombre'];  $sql = "UPDATE personal SET pers\_nombre = '$nombre' WHERE pers\_id = $id";  if ($bd->query($sql) === TRUE) {  header('Location: index.php');  } else {  echo "Error: " . $sql . " " . $bd->error;  }  }  ?>  <form method="post">  <label for="id">ID</label>  <input required name="id" class="form-input" type="text" id="id">  <label for="nombre">Nombre</label>  <input required name="nombre" class="form-input" type="text" id="nombre">  <button type="submit" value="Modificar">Modificar</button>  </form> | |  | | | **Tabla** | <?php include 'conexion.php';  $bd = conectarBD();  $result = $bd->query("SELECT \* FROM personal");  if ($result->num\_rows > 0) {  echo "<table><tr><th>ID</th><th>Nombre</th></tr>";  while($row = $result->fetch\_assoc()) {  echo "<tr><td>" . $row["pers\_id"]. "</td><td>" . $row["pers\_nombre"]. "</td></tr>";  }  echo "</table>";  } else {  echo "0 resultados";  }  ?> | |  | | | **Buscar (Señala ID)** | <?php include 'conexion.php';  $conexion = conectarBD();  $sql\_todos = "SELECT \* FROM personal";  $resultado\_todos = mysqli\_query($conexion, $sql\_todos);  if (isset($\_GET['id'])) {  $id = $\_GET['id'];  $consulta = "SELECT \* FROM personal WHERE pers\_idpersonal = '$id'";  $resultado = mysqli\_query($conexion, $consulta);  if ($resultado && mysqli\_num\_rows($resultado) > 0) {  $row = mysqli\_fetch\_assoc($resultado);  } else {  echo "<script>alert('No se encontró ningún registro con el ID proporcionado.');</script>";  }  }?>  <div>  <form method="GET">  <label for="id">ID del Personal:</label>  <input type="text" id="id" name="id" placeholder="Ingrese ID del Personal" required>  <button type="submit">Buscar</button>  </form>  </div>  <div>  <table border="1">  <tr>  <th>ID</th>  <th>Nombre</th>  </tr>  <?php  if ($resultado\_todos->num\_rows > 0) {  while ($row = $resultado\_todos->fetch\_assoc()) {  echo "<tr";  if (isset($\_GET['id']) && $row["pers\_idpersonal"] == $\_GET['id']) {  echo " style='background-color: yellow;'";  }  echo ">";  echo "<td>" . $row["pers\_idpersonal"] . "</td>";  echo "<td>" . $row["pers\_nombre"] . "</td>"; echo "</tr>";  }  } else {  echo "<tr><td colspan='2'>No hay datos del personal</td></tr>";  }  ?>  </table>  </div> |   **Entregar**:  • Agregar su firma digital en la primera hoja.  • Subir al aula virtual, este documento con su desarrollo, los archivos codificados y el archivo SQL de la base de datos. |