**Conocimientos SQL**

1. Utilice las siguientes tablas, construir las consultas que se solicitan a en los ítems 2, 3 y 4.

**Tabla**: **PRUEBAESTUDIANTE**

|  |  |  |  |
| --- | --- | --- | --- |
| Eid | Nombre | Especialidad | Grado |
| 100 | JONES | HISTORIA | GR |
| 150 | PARKS | CONTABILIDAD | SO |
| 200 | BAKER | MATEMATICAS | GR |
| 250 | GLASS | HISTORIA | SN |
| 300 | BAKER | CONTABILIDAD | SN |
| 350 | RUSSELL | MATEMATICAS | JR |
| 400 | REY | CONTABILIDAD | FR |
| 450 | JONES | HISTORIA | SN |

**Tabla**: **PRUEBAINSCRIPCION**

|  |  |  |
| --- | --- | --- |
| Eid | NombreClase | Posicion |
| 100 | BD445 | 1 |
| 150 | BA200 | 1 |
| 200 | BD445 | 2 |
| 200 | CS250 | 1 |
| 300 | CS150 | 1 |
| 400 | BA200 | 2 |
| 400 | BF410 | 1 |
| 400 | CS250 | 2 |
| 450 | BA200 | 3 |

**Tabla**: **PRUEBACLASE**

|  |  |  |
| --- | --- | --- |
| Nombre | Horario | Aula |
| BA200 | M-F9 | SC110 |
| BD445 | MWF3 | SC213 |
| BF410 | MWF8 | SC213 |
| CS150 | MWF3 | EA304 |
| CS250 | MWF12 | EB210 |

Construir el Query que permita obtener los siguientes datos:

1. De la tabla **PRUEBAESTUDIANTE** se desea saber el nombre de las especialidades registradas y la cantidad de veces que existe en la tabla.

Resultado esperado:

|  |  |
| --- | --- |
| CONTABILIDAD | 3 |
| HISTORIA | 3 |
| MATEMATICAS | 2 |

R://

**Select COUNT(pe.Eid), pe.Especialidad FROM PRUEBAESTUDIANTE pe GROUP BY pe.Especialidad;**

1. Se requiere consultar cuales son las inscripciones que tienen horario **M-F9.**  Resultado esperado:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Eid | NombreClase | Posicion | Nombre | Horario | Aula |
| 150 | BA200 | 1 | BA200 | M-F9 | SC110 |
| 400 | BA200 | 2 | BA200 | M-F9 | SC110 |
| 450 | BA200 | 3 | BA200 | M-F9 | SC110 |

R://

**Select pi.\*, pc.\* FROM PRUEBAINSCRIPCION pi INNER JOIN PRUEBACLASE pc ON pi.NombreClase = pc.Nombre where pc.Horario = ‘M-F9’**

1. Se ha solicitado un reporte donde se indiquen los datos de los estudiantes que no han realizado su proceso de inscripción. Resultado esperado.

|  |  |  |
| --- | --- | --- |
| Eid | nombre | grado |
| 250 | GLASS | SN |
| 350 | RUSSELL | JR |

R://

**SELECT pe.Eid, pe.Nombre, pe.Grado**

**FROM PRUEBAESTUDIANTE pe**

**WHERE pe.Eid NOT IN (select pi.Eid FROM PRUEBAINSCRIPCION));**

1. Se tienen un archivo en texto plano con más de 5000 registros (filas) y 20 columnas. Se desea incluir esta información en una tabla de la base de datos que ya existe con el objetivo de iniciar un nuevo proceso. Brevemente describa de qué manera realizaría este proceso y estime el tiempo que invierte en realizarlo.

R://

**Crearía un store procedure para la migración de datos y desde Backend puro haría el script para consultar el texto plano de forma asíncrona y así ir consumiendo el procedure ya que puede que la transacción falle por lo cual se manejaría por lotes y no saturar el servidor de la base de datos, el tiempo estimado seria de 5 a 6h dependiendo de los errores que puedan suceder al migrar los datos.**

**Prueba Técnica**

1. Se debe crear una solución web en la que se puedan ver los estudiantes inscritos en una academia de conducción con las clases que tengan asignadas. El objetivo de esta prueba es crear una base de datos dónde alojar la información pertinente del estudiante. Desde la interfaz gráfica poder crear los estudiantes, mostrar el listado de estos y tener la opción de ver el detalle (información básica, módulos inscritos y clases activas). Se debe manejar independiente el desarrollo a nivel de Front y Back, teniendo en cuenta los siguientes criterios:
   1. Se requiere guardar la información de los estudiantes de una academia de conducción, donde del estudiante se guarde: el nombre, la edad y el documento de identidad y el tipo de licencia (A1, A2, B1, B2, C1, C2).
   2. Las clases están dividas por módulos (Adaptación, Ética, Marco Legal) y cada módulo cuenta con clases (Adaptación 1, Adaptación 2, Ética 1, Ética 2). El estudiante no puede estar activo en más de un nivel del mismo módulo.
2. El desarrollo a nivel de back debe estar sobre servicios RestApi C# .Net o java
3. El desarrollo a nivel de Front lo ideal es que sea sobre Angular, ya que el framework con el que actualmente se está trabajando en telefónica y con que el que se tendría que llegar a trabajar.
4. El mínimo producto que se debe entregar es a nivel de Front, se pueda visualizar el listado de los estudiantes, y la creación de estos, teniendo en cuenta validaciones.

Tener en cuenta:

* Buenas prácticas de programación.
* Validaciones de campos.

Entregables:

* Código Back
* Código Front
* Script de la base de datos creada e Insert usados para poblar la base de datos.