



**GENERALITAT
VALENCIANA**

Conselleria de Justícia
y Administración Pública

**CONVOCATORIA PRUEBAS SELECTIVAS DE ACCESO
AL CUERPO B-01. TÉCNICO DE GESTIÓN DE
SISTEMAS Y TECNOLOGÍAS DE LA INFORMACIÓN DE
LA ADMINISTRACIÓN**

TURNO LIBRE

CONVOCATORIA 62/25

SEGUNDO EJERCICIO

TIEMPO REALIZACIÓN:

(3 horas)

Atención, seleccione la opción A o B para la realización del examen.

Opción A

1. Diagrama entidad relación y diseño de tablas. Sistema de Gestión Tributaria (12 puntos).

Se desea modelar un sistema completo de gestión de impuestos para una administración pública.

Requisitos

1. Contribuyentes

El sistema gestiona contribuyentes:

Atributos: NIF, nombre, dirección, email.

Tipos:

Persona Física (tiene además estado civil)

Empresa (tiene razón social y número de empleados)

Un contribuyente puede tener representantes legales:

Un representante también es un contribuyente.

Un contribuyente puede tener varios representantes.

2. Declaraciones

Los contribuyentes presentan declaraciones tributarias:

Atributos: número, fecha, estado (borrador, presentada, validada), importe total.

Cada declaración pertenece a un único contribuyente.

Un contribuyente puede presentar varias declaraciones.

Cada declaración corresponde a un ejercicio fiscal:

Atributos: año, fecha inicio, fecha fin.

3. Impuestos y Detalles

Cada declaración incluye varios impuestos:

Atributos: nombre, tipo (IVA, IRPF, Sociedades), porcentaje.

Relación:

Una declaración puede tener varios impuestos.

Un impuesto puede estar en varias declaraciones.

Se debe almacenar:

Base imponible

Cuota calculada

4. Pagos

Las declaraciones pueden pagarse mediante uno o varios pagos:

Atributos: fecha, importe, método (transferencia, tarjeta, domiciliación).

Condiciones:

- Una declaración puede tener varios pagos.
- Puede haber pagos parciales.

5. Sanciones

Si una declaración se presenta fuera de plazo:

Se genera una sanción:

- Atributos: importe, motivo, fecha.
- Una declaración puede tener 0 o 1 sanción.

6. Inspecciones

El sistema gestiona inspecciones tributarias:

- Atributos: número, fecha inicio, fecha fin, resultado.
- Una inspección puede afectar a varias declaraciones.
- Una declaración puede estar en varias inspecciones.

Se solicita

- 1.A. Identifica entidades (2 puntos).
- 1.B. Define atributos (2 puntos).
- 1.C. Establece Relaciones y Multiplicidades (3 puntos).
- 1.D. Diagrama de Entidad-Relación (2 puntos).
- 1.E. Script sql de diseño de tablas (3 puntos).

2. Programación Java (7 puntos).

Se desea desarrollar un programa en Java para gestionar pacientes de un centro sanitario.

De cada paciente se desea almacenar:

- NIF
- Nombre
- Edad
- Número de visitas realizadas

El programa debe permitir:

- 2.A. Crear una clase Paciente con sus atributos (1,5 puntos).
- 2.B. Incrementar el número de visitas de un paciente (1 punto).
- 2.C. Calcular si un paciente es mayor de edad (1,5 puntos).
- 2.D. Mostrar los datos del paciente (1,5 puntos).
- 2.E. En el programa principal (1,5 puntos):
 - Guardar varios pacientes en un ArrayList
 - Mostrar los datos de todos los pacientes
 - Mostrar cuántos pacientes hay en total

3.- SQL (8 puntos)

La empresa TransLogiX gestiona empleados, vehículos, rutas y entregas. A partir de las siguientes tablas, responde a las cuestiones planteadas utilizando sentencias SQL.

Tabla Empleados

- id_empleado (INT, PK)
- nombre (VARCHAR)

- puesto (VARCHAR)
- salario (DECIMAL)

Tabla Vehiculos

- id_vehiculo (INT, PK)
- matricula (VARCHAR)
- tipo (VARCHAR)
- capacidad(INT)

Tabla Rutas

- id_ruta (INT, PK)
- origen (VARCHAR)
- destino(VARCHAR)
- distancia(INT)

Tabla Entregas

- id_entrega (INT, PK)
- id_empleado (INT)
- id_vehiculo (INT)
- id_ruta(INT)
- fecha (DATE)
- peso (INT)

3.A. Obtener el nombre y puesto de todos los empleados cuyo puesto sea *Conductor* (0,5 puntos).

3.B. Mostrar los vehículos con capacidad superior a 1.000 kg (0,5 puntos).

3.C. Listar todas las entregas mostrando: nombre del empleado, matrícula del vehículo y destino de la ruta (1 punto).

3.D. Mostrar todas las entregas realizadas por la empleada Ana Gómez (1 punto).

3.E. Calcular el número total de entregas realizadas por cada empleado (1 punto).

3.F. Obtener la carga media transportada por cada tipo de vehículo (1 punto).

3.G. Mostrar los empleados cuyo salario sea superior al salario medio de la empresa (1,5 puntos).

3.H. Obtener las rutas utilizadas en entregas cuyo peso supere los 1.000 kg (0,5 puntos).

3.I. Aumentar en 100 € el salario de todos los empleados cuyo puesto sea *Conductor* (0,5 puntos).

3.J. Eliminar todas las entregas realizadas antes del 15 de enero de 2024 (0,5 puntos).

4.- Seguridad (3 puntos): Explica las principales medidas de seguridad en bases de datos.

Control de acceso (usuarios y roles), Privilegios (SELECT, INSERT, UPDATE, DELETE), Copias de seguridad (backups), Protección frente a inyección SQL, Otro.

Opción B

1. Diagrama de clases. Sistema de Gestión Integral de Vehículos (12 puntos).

Se desea modelar un sistema completo de gestión de vehículos.

1. Propietarios

El sistema gestiona propietarios de vehículos: Atributos: NIF, nombre, dirección, email.

Tipos:

- Persona Física (incluye además *estado civil*)
- Empresa (incluye *razón social* y *número de empleados*)

Un propietario puede tener representantes legales:

- Un representante también es un propietario.
- Un propietario puede tener varios representantes.

2. Vehículos

Los propietarios registran vehículos en el sistema: Atributos: matrícula, marca, modelo, año, estado (activo, baja temporal, baja definitiva).

- Cada vehículo pertenece a un único propietario.
- Un propietario puede tener varios vehículos.

Cada vehículo está asociado a un permiso de circulación: Atributos: número, fecha expedición, fecha caducidad.

3. Impuestos y Tasas

Cada vehículo debe pagar impuestos periódicos: Atributos: nombre, tipo (circulación, matriculación, emisiones), porcentaje o tarifa.

Relación:

- Un vehículo puede tener varios impuestos asociados.
- Un impuesto puede aplicarse a varios vehículos.

Se debe almacenar:

- Base imponible
- Cuota calculada

4. Pagos

Los impuestos pueden pagarse mediante uno o varios pagos: Atributos: fecha, importe, método (transferencia, tarjeta, domiciliación).

Condiciones:

- Un impuesto puede tener varios pagos.
- Puede haber pagos parciales.

5. Sanciones

Si un vehículo incumple alguna obligación (ITV caducada, impago, estacionamiento indebido...):

- Se genera una sanción: Atributos: importe, motivo, fecha.
- Un vehículo puede tener 0 o 1 sanción por infracción concreta.

6. Inspecciones

El sistema gestiona inspecciones técnicas (ITV) o inspecciones extraordinarias:
Atributos: fecha inicio, fecha fin, resultado.

Relación:

- Una inspección puede afectar a varios vehículos.
- Un vehículo puede estar en varias inspecciones.

Se solicita

- 1.A. Identifica las clases (2 puntos).
- 1.B. Define: (3 puntos).
 - Atributos
 - Métodos principales
- 1.C. Establece: (3,5 puntos).
 - Herencia
 - Asociaciones
 - Agregación o composición (si procede)
- 1.D. Indica multiplicidades: (2 puntos).
 - 1, 0.., 1.., etc.
- 1.E. Diagrama de clases. (1,5 puntos).

2. Lenguajes de Marcas – XSLT (8 puntos).

Dado el xml siguiente.

```
<?xml version="1.0" encoding="UTF-8"?>
<centroSalud>
  <paciente nif="123A">
    <nombre>Juan</nombre>
    <edad>30</edad>
    <visitas>3</visitas>
  </paciente>

  <paciente nif="456B">
    <nombre>María</nombre>
    <edad>16</edad>
    <visitas>1</visitas>
  </paciente>

  <paciente nif="789C">
    <nombre>Luis</nombre>
    <edad>45</edad>
    <visitas>5</visitas>
  </paciente>
</centroSalud>
```

2.A.- Codifica la página html que retorna este xslt (1 punto).

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

  <!-- Plantilla raíz -->
  <xsl:template match="/centroSalud">
    <html>
```

```

<head>
  <title>Listado de pacientes</title>
  <meta charset="UTF-8"/>
</head>
<body>
  <h1>Listado de pacientes</h1>
  <ul>
    <!-- Recorremos cada paciente -->
    <xsl:for-each select="paciente">
      <li>
        <xsl:value-of select="nombre"/>
      </li>
    </xsl:for-each>
  </ul>
</body>
</html>
</xsl:template>

```

2.B. Construye un xslt que muestre el nombre y número de visitas en un HTML (1,5 puntos).

2.C. Construye un xslt que muestre una tabla HTML con nombre y visitas de cada paciente. (2 puntos).

2.D. Construye un xslt que muestra una frase por cada paciente con el siguiente formato: (2 puntos).

Paciente Juan – NIF: 123A

2.E. Explica qué hace la instrucción xsl:for-each y para qué se utiliza. (1,5 puntos).

3. SQL (7 puntos)

Sea la siguiente base de datos:

Tabla alumnos

- id_alumno (INT, PK)
- nombre (VARCHAR)
- edad (INT)

Tabla modulos

- id_modulo (INT, PK)
- nombre (VARCHAR)

Tabla matriculas

- id_alumno (INT, FK)
- id_modulo (INT, FK)
- nota (DECIMAL)

Construye las sentencias sql que:

3.A. Muestra el nombre de todos los alumnos mayores de 20 años (0,75 puntos).

3.B. Obtén el nombre de los módulos donde hay alumnos con nota superior a 8 (0,75 puntos).

3.C. Muestra el nombre de cada alumno y su nota media (0,9 puntos).

3.D. Obtén los alumnos que no están matriculados en ningún módulo (0,9 puntos).

3.E. Muestra el alumno con la nota más alta de toda la base de datos (1 punto).

3.F. Obtén el número de alumnos matriculados en cada módulo (0,9 puntos).

3.G. Muestra los alumnos que tienen más de una matrícula (0,9 puntos).

3.H. Obtén el nombre de los alumnos cuya nota media es superior a la media de todos los alumnos (0,9 puntos).

4.- Seguridad: Explica los principios básicos de seguridad en aplicaciones informáticas (3 puntos). Autenticación vs autorización, Confidencialidad, integridad y disponibilidad, Cifrado (simétrico vs asimétrico), Buenas prácticas en desarrollo seguro, Otros.

**CONVOCATÒRIA PROVES SELECTIVES
D'ACCÉS AL COS B-01. TÈCNIC DE GESTIÓ
DE SISTEMES I TECNOLOGIES DE LA
INFORMACIÓ DE L'ADMINISTRACIÓ**

TORN LLIURE

CONVOCATÒRIA 62/25

SEGON EXERCICI

TEMPS REALITZACIÓ:

(3 hores)

Atenció, seleccione l'opció A o B per a la realització de l'examen.

Opció A

1. Diagrama entitat relació i disseny de taules. Sistema de Gestió Tributària (12 punts).

Es vol modelar un sistema complet de gestió d'impostos per a una administració pública.

Requisits

1. Contribuents

El sistema gestiona contribuents:

Atributs: NIF, nom, adreça, correu electrònic.

Tipus:

Persona física (té, a més, estat civil)

Empresa (té raó social i nombre d'empleats)

Un contribuent pot tindre representants legals:

Un representant també és un contribuent.

Un contribuent pot tindre diversos representants.

2. Declaracions

Els contribuents presenten declaracions tributàries:

Atributs: número, data, estat (esborrany, presentada, validada), import total.

Cada declaració pertany a un únic contribuent.

Un contribuent pot presentar diverses declaracions.

Cada declaració correspon a un exercici fiscal:

Atributs: any, data inici, data fi.

3. Impostos i detalls

Cada declaració inclou diversos impostos:

Atributs: nom, tipus (IVA, IRPF, societats), percentatge.

Relació:

Una declaració pot tindre diversos impostos.

Un impost pot estar en diverses declaracions.

S'ha d'emmagatzemar:

Base imposable

Quota calculada

4. Pagaments

Les declaracions poden pagar-se mitjançant un o diversos pagaments:

Atributs: data, import, mètode (transferència, targeta, domiciliació).

Condicions:

Una declaració pot tindre diversos pagaments.
Poden haver-hi pagaments parcials.

5. Sancions

Si una declaració es presenta fora de termini:

Es genera una sanció:

Atributs: import, motiu, data.

Una declaració pot tindre 0 o 1 sanció.

6. Inspeccions

El sistema gestiona inspeccions tributàries:

Atributs: número, data inici, data fi, resultat.

Una inspecció pot afectar diverses declaracions.

Una declaració pot estar en diverses inspeccions.

Se sol·licita

1.A. Identifica entitats (2 punts).

1.B. Definix atributs (2 punts).

1.C. Establix relacions i multiplicitats (3 punts).

1.D. Diagrama d'entitat-relació (2 punts).

1.E. Script sql de disseny de taules (3 punts).

2. Programació Java (7 punts).

Es vol desenrotllar un programa en Java per a gestionar pacients d'un centre sanitari.

De cada pacient es vol emmagatzemar:

NIF

Nom

Edat

Nombre de visites realitzades

El programa ha de permetre:

2.A. Crear una classe Pacient amb els seus atributs (1,5 punts).

2.B. Incrementar el nombre de visites d'un pacient (1 punt).

2.C. Calcular si un pacient és major d'edat (1,5 punts).

2.D. Mostrar les dades del pacient (1,5 punts).

2.E. En el programa principal (1,5 punts):

- Guardar diversos pacients en un ArrayList
- Mostrar les dades de tots els pacients
- Mostrar quants pacients hi ha en total

3.- SQL (8 punts)

L'empresa TransLogiX gestiona empleats, vehicles, rutes i lliuraments. A partir de les taules següents, respon a les qüestions plantejades utilitzant sentències SQL.

Taula empleats

- id_empleat (INT, PK)
- nom (VARCHAR)
- lloc (VARCHAR)
- salari (DECIMAL)

Taula vehicles

- id_vehicle (INT, PK)
- matrícula (VARCHAR)
- tipus (VARCHAR)
- capacitat (INT)

Taula rutes

- id_ruta (INT, PK)
- origen (VARCHAR)
- destí (VARCHAR)
- distància (INT)

Taula lliuraments

- id_entrega (INT, PK)
- id_empleat (INT)
- id_vehicle (INT)
- id_ruta (INT)
- data (DATE)
- pes (INT)

3.A. Obtindre el nom i el lloc de tots els empleats el lloc dels quals siga *Conductor* (0,5 punts).

3.B. Mostrar els vehicles amb capacitat superior a 1.000 kg (0,5 punts).

3.C. Llistar tots els lliuraments mostrant: nom de l'empleat, matrícula del vehicle i destí de la ruta (1 punt).

3.D. Mostrar tots els lliuraments realitzats per l'empleada Anna Gómez (1 punt).

3.E. Calcular el nombre total de lliuraments realitzats per cada empleat (1 punt).

3.F. Obtindre la càrrega mitjana transportada per cada tipus de vehicle (1 punt).

3.G. Mostrar els empleats el salari dels quals siga superior al salari mitjà de l'empresa (1,5 punts).

3.H. Obtindre les rutes utilitzades en entregues el pes de les quals supere els 1.000 kg (0,5 punts).

3.I. Augmentar en 100 € el salari de tots els empleats el lloc dels quals siga *Conductor* (0,5 punts).

3.J. Eliminar tots els lliuraments realitzats abans del 15 de gener de 2024 (0,5 punts).

4.- Seguretat: Explica les principals mesures de seguretat en bases de dades. Control d'accés (usuaris i rols), privilegis (SELECT, INSERT, UPDATE, DELETE), Còpies de seguretat (*backups*), protecció davant d'injecció SQL, altres (3 punts).

Opció B

1. Diagrama de classes. Sistema de gestió integral de vehicles (12 punts).

Es desitja modelar un sistema complet de gestió de vehicles per a una administració pública.

1. Propietaris

El sistema gestiona propietaris de vehicles: Atributs: NIF, nom, direcció, correu electrònic.

Tipus:

- Persona física (inclou a més *estat civil*)
- Empresa (inclou *raó social* i *nombre d'empleats*)

Un propietari pot tindre representants legals:

- Un representant també és un propietari.
- Un propietari pot tindre diversos representants.

2. Vehicles

Els propietaris registren vehicles en el sistema: Atributs: matrícula, marca, model, any, estat (actiu, baixa temporal, baixa definitiva).

- Cada vehicle pertany a un únic propietari.
- Un propietari pot tindre diversos vehicles.

Cada vehicle està associat a un permís de circulació: Atributs: número, data expedició, data caducitat.

3. Impostos i taxes

Cada vehicle ha de pagar impostos periòdics: Atributs: nom, tipus (circulació, matriculació, emissions), percentatge o tarifa.

Relació:

- Un vehicle pot tindre diversos impostos associats.
- Un impost pot aplicar-se a diversos vehicles.

S'ha d'emmagatzemar:

- Base imposable
- Quota calculada

4. Pagaments

Els impostos poden pagar-se mitjançant un o diversos pagaments: Atributs: data, import, mètode (transferència, targeta, domiciliació).

Condicions:

- Un impost pot tindre diversos pagaments.
- Pot haver-hi pagaments parcials.

5. Sancions

Si un vehicle incomplix alguna obligació (ITV caducada, impagament, estacionament indegut...):

- Es genera una sanció: Atributs: import, motiu, data.
- Un vehicle pot tindre 0 o 1 sanció per infracció concreta.

6. Inspeccions

El sistema gestiona inspeccions tècniques (ITV) o inspeccions extraordinàries: Atributs: data inici, data fi, resultat.

Relació:

- Una inspecció pot afectar diversos vehicles.
- Un vehicle pot estar en diverses inspeccions.

Se sol·licita

- 1.A. Identifica les classes (2 punts).
- 1.B. Definix: (3 punts).
 - Atributs
 - Mètodes principals
- 1.C. Establix: (3,5 punts).
 - Herència
 - Associacions
 - Agregació o composició (si és procedent)
- 1.D. Indica multiplicitats: (2 punts).
 - 1, 0.., 1.., etc.
- 1.E. Diagrama de classes. (1,5 punts).

2. Llenguatges de marques – XSLT (8 punts).

Atés l'xml següent.

```
<?xml version="1.0" encoding="UTF-8"?>
<centroSalud>
  <paciente nif="123A">
    <nombre>Juan</nombre>
    <edad>30</edad>
    <visitas>3</visitas>
  </paciente>

  <paciente nif="456B">
    <nombre>María</nombre>
    <edad>16</edad>
    <visitas>1</visitas>
  </paciente>

  <paciente nif="789C">
    <nombre>Luis</nombre>
    <edad>45</edad>
    <visitas>5</visitas>
  </paciente>
</centroSalud>
```

2.A.- Codifica la pàgina html que retorna este xslt (1 punt).

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

  <!-- Plantilla raíz -->
  <xsl:template match="/centroSalud">
    <html>
      <head>
```

```

<title>Listado de pacientes</title>
<meta charset="UTF-8"/>
</head>
<body>
  <h1>Listado de pacientes</h1>
  <ul>
    <!-- Recorremos cada paciente -->
    <xsl:for-each select="paciente">
      <li>
        <xsl:value-of select="nombre"/>
      </li>
    </xsl:for-each>
  </ul>
</body>
</html>
</xsl:template>

```

- 2.B. Construeix un xslt que mostre el nom i nombre de visites en un HTML (1,5 punts).
- 2.C. Construeix un xslt que mostre una taula HTML amb nom i visites de cada pacient. (2 punts).
- 2.D. Construeix un xslt que mostre una frase per cada pacient amb el següent format: (2 punts).
Pacient Juan – NIF: 123A
- 2.E. Explica què fa la instrucció xsl:for-each i per a què s'utilitza. (1,5 punts).

3. SQL (7 punts)

Amb la següent base de dades:

Taula alumnes

- id_alumne (INT, PK)
- nom (VARCHAR)
- edat (INT)

Taula mòduls

- id_mòdul (INT, PK)
- nom (VARCHAR)

Taula matrícules

- id_alumne (INT, FK)
- id_mòdul (INT, FK)
- nota (DECIMAL)

Construeix les sentències sql que:

- 3.A. Mostra el nom de tots els alumnes majors de 20 anys (0,75 punts).
- 3.B. Obtén el nom dels mòduls on hi ha alumnes amb nota superior a 8 (0,75 punts).
- 3.C. Mostra el nom de cada alumne i la seua nota mitjana (0,9 punts).
- 3.D. Obtén els alumnes que no estan matriculats en cap mòdul (0,9 punts).
- 3.E. Mostra l'alumne amb la nota més alta de tota la base de dades (1 punt).
- 3.F. Obtén el nombre d'alumnes matriculats en cada mòdul (0,9 punts).
- 3.G. Mostra els alumnes que tenen més d'una matrícula (0,9 punts).
- 3.H. Obtén el nom dels alumnes la nota mitjana dels quals és superior a la mitjana de tots els alumnes (0,9 punts).

4. Seguretat: Explica els principis bàsics de seguretat en aplicacions informàtiques (3 punts). Autenticació vs autorització, Confidencialitat, integritat i disponibilitat, Xifrat (simètric vs asimètric), Bones pràctiques en desenrotllament segur, Altres.