

AYUNTAMIENTO DE ALICANTE

ANUNCIO PRUEBAS DE SELECCIÓN DE PERSONAL

CONVOCATORIA N.º 513

ANUNCIO N.º 4

PLAZAS: BOLSA DE TÉCNICO/A SUPERIOR EN INFORMÁTICA
TURNOS: LIBRE

ASUNTO: Resultados 2º ejercicio y resultados provisionales por orden de puntuación.

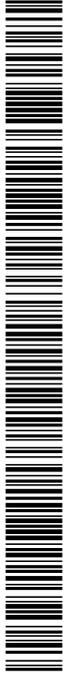
El Tribunal Calificador, por unanimidad, ha acordado:

PRIMERO.- Hacer pública la relación provisional de aspirantes que han superado el segundo ejercicio del proceso selectivo, con las siguientes calificaciones:

Opositor	DNI-NIE	Nombre	CALIFICACIÓN
1	***0911**	RODRIGUEZ MORALA, JORGE	7,5
2	***2540**	RUIZ GUILLO, DAVID	9
3	***5196**	RUIZ PIÑA, MARIA ANGELES	5

SEGUNDO: Hacer público el supuesto y los criterios de corrección del segundo ejercicio (ANEXO).

TERCERO.- Hacer públicos los resultados provisionales totales obtenidos por los aspirantes que han superado la totalidad de los ejercicios del proceso selectivo, con el orden en que quedan clasificados:



N.º ORDEN	DNI-NIE	Nombre	1º EJERCICIO	2º EJERCICIO	TOTAL
1	***2540**	RUIZ GUILLO, DAVID	7,01	9	16,01
2	***0911**	RODRIGUEZ MORALA, JORGE	6,27	7,5	13,77
3	***5196**	RUIZ PIÑA, MARIA ANGELES	5,21	5	10,21

CUARTO.- Conceder a los aspirantes que deseen revisar su segundo ejercicio el plazo de TRES DÍAS HÁBILES desde la publicación del presente Anuncio para solicitar dicha revisión.

INSTRUCCIONES PARA SOLICITAR LA REVISIÓN: El aspirante que desee revisar su ejercicio deberá dirigir un correo electrónico a la dirección dolores.alcaraz@alicante.es indicando en el Asunto: "CONVOCATORIA 513 BOLSA TÉCNICO SUPERIOR INFORMÁTICA", debiendo identificarse mediante nombre, apellidos y DNI, solicitando expresamente la revisión del segundo ejercicio.

Se hace constar expresamente que dicha dirección de correo únicamente está habilitada para recepcionar las solicitudes de revisión, sin que sea medio apto ni válido para el resto de comunicaciones que los aspirantes deseen dirigir al Tribunal Calificador.

QUINTO.- La revisión tendrá lugar el próximo **29 DE ABRIL DE 2025 a las 9:00 horas**, en las Dependencias del Servicio de Innovación, Informática y Agenda Digital, sito en la calle Jorge Juan n.º 4, 4ª planta, para aquellos aspirantes que previamente lo hayan solicitado.

SEXTO.- Se concede un plazo de 5 días hábiles para que los aspirantes puedan presentar alegaciones a las puntuaciones otorgadas, finalizado el cual y, en su caso, resueltas las alegaciones presentadas, se hará público el listado de las puntuaciones definitivas del proceso selectivo.

Lo que se hace público a los debidos efectos.

La Secretaria del Tribunal,

Dolores María Alcaraz Jerez
(documento firmado electrónicamente)



El Ayuntamiento pretende informatizar las bibliotecas municipales. De la entrevista con el personal de las bibliotecas, se han obtenido los siguientes requisitos:

1. Cada socio que emplea la biblioteca tiene un carnet, con un número de socio que es único, y nos gustaría almacenar, su nombre, apellidos, teléfono y un correo electrónico. Un socio, puede sacar libros en cualquier biblioteca de la ciudad. De las bibliotecas nos gustaría almacenar su nombre y dirección.
2. Sobre un libro, se tienen varios ejemplares distribuidos por las bibliotecas. Algunos ejemplares, solo pueden ser consultados en la biblioteca y otros pueden ser prestados a los socios. Los ejemplares que se prestan nos interesan saber a quién se ha prestado, la fecha del préstamo y la fecha de devolución. Sobre cada libro es necesario saber su título, ISBN y los autores. Un libro puede tener varios autores y un autor puede escribir muchos libros. Sobre cada ejemplar nos interesa saber la fecha de compra, precio y un campo notas. Sobre el autor nos interesa saber su nombre y nacionalidad.
3. A la aplicación entraran los usuarios usubi1, usubi2, usubi3, .. usubiN.
4. Todos los usuarios de la aplicación podrán consultar y actualizar información almacenada sobre los socios. Solo los usuarios que los clasifiquemos como gestores podrán consultar, actualizar y añadir nuevos socios. Solo los usuarios que clasifiquemos como administradores de la aplicación podrán consultar, actualizar, añadir y borrar socios.
5. Se requiere guardar información sobre todas las actualizaciones que se realiza sobre la información del socio, en concreto se necesita saber: quien ha realizado la operación, cuando, y toda la información que contenía el registro antes de la modificación.

Se solicita:

1. Dibuja el diagrama entidad-relación (5 puntos)
2. Teniendo en cuenta que el Ayuntamiento trabaja con el SGBD Oracle.
 - a. Escribe los comandos SQL para la creación de las tablas del esquema anterior (2 puntos),
 - b. Explica y escribe el comando SQL para dar respuesta al requisito 3 (1 puntos),
 - c. Explica y escribe el comando SQL para dar respuesta al requisito 4 (1 puntos),
 - d. Explica y escribe el comando SQL para dar respuesta al requisito 5 (1 puntos).

Código Seguro de Verificación: 6e7b1447-24dc-4533-9adc-
f6730aa77b4d
Origen: Administración
Identificador documento original: ES_L01030149_2025_22087989
Fecha de impresión: 23/04/2025 12:28:56
Página 4 de 9

FIRMAS
1.- DOLORES M ALCARAZ JEREZ (Secretaria del Tribunal), 23/04/2025 12:28

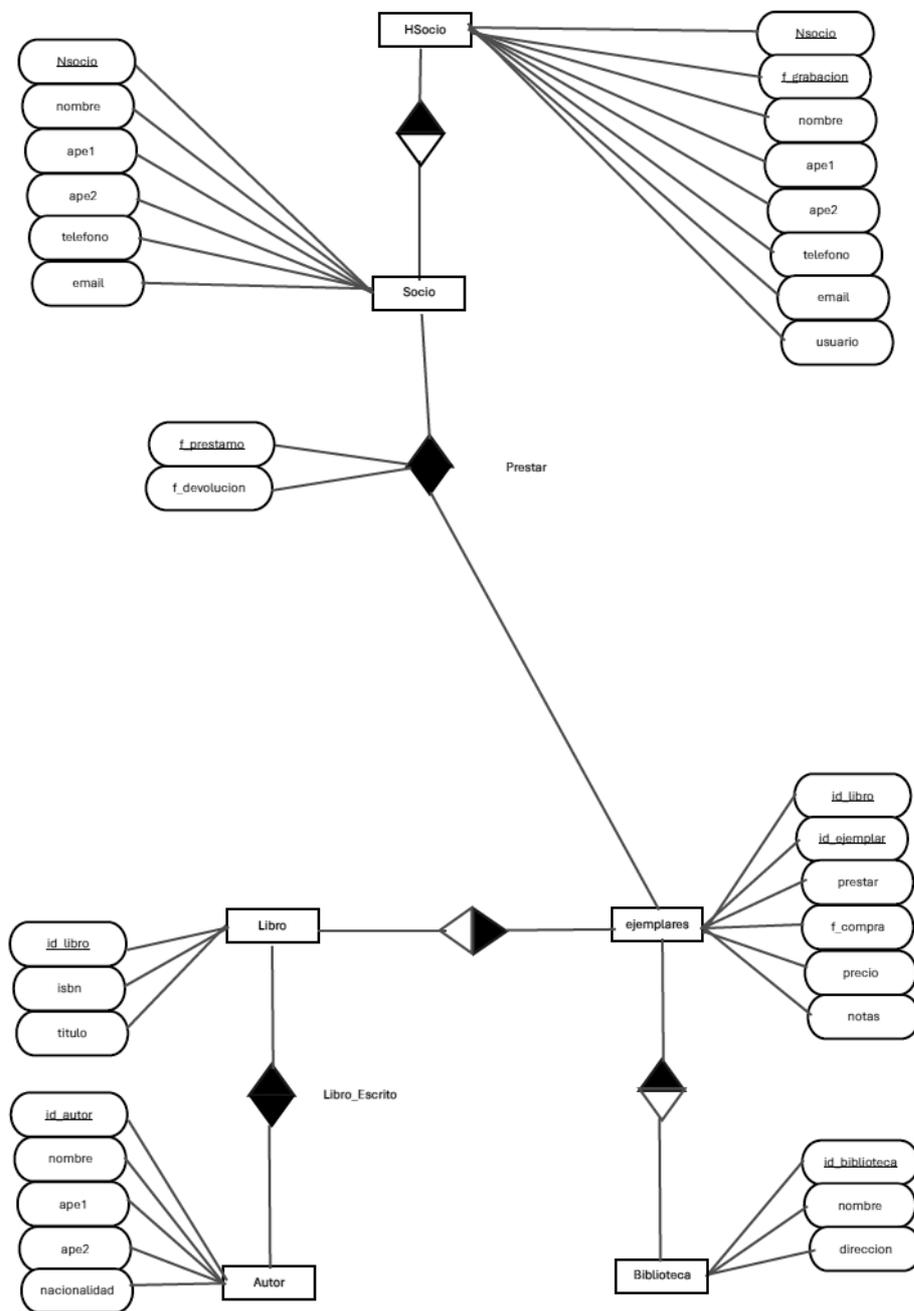
Respuestas.

Una de las posibles soluciones al enunciado.





Respuesta 1 a





Respuesta 2 a

```
create table propietario.socio (
```

```
nsocio      number( 5 ) not null,
```

```
nombre      varchar2(50) not null ,
```

```
ape1        varchar2(50) not null ,
```

```
ape2        varchar2(50) null ,
```

```
teléfono    varchar2(20) null ,
```

```
email       varchar2(20) null ,
```

```
constraint pk_socio primary key (nsocio ) );
```

```
create table propietario.hsocio (
```

```
nsocio      number( 5 ) not null,
```

```
nombre      varchar2(50) not null ,
```

```
ape1        varchar2(50) not null ,
```

```
ape2        varchar2(50) null ,
```

```
teléfono    varchar2(20) null ,
```

```
email       varchar2(20) null ,
```

```
usuario     varchar2(10) DEFAULT substr(user,1,10) not null,
```

```
f_grabacion date DEFAULT sysdate not null,
```

```
constraint pk_hsocio primary key (nsocio, f_grabacion ),
```

```
constraint fk_hsocio_socio foreign key ( nsocio ) references propietario.socio ( nsocio ) );
```

```
create table propietario.autor (
```

```
id_autor    number( 5 ) not null,
```

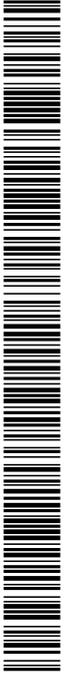
```
nombre      varchar2(50) not null ,
```

```
ape1        varchar2(50) not null ,
```

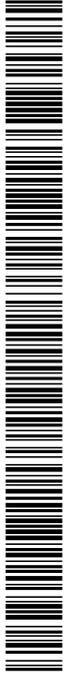
```
ape2        varchar2(50) null ,
```

```
nacionalidad varchar2(20) null ,
```

```
constraint pk_autor primary key ( id_autor ) );
```



```
create table propietario.libro (  
id_libro      number( 5 ) not null,  
titulo        varchar2(50) not null ,  
ISBN          varchar2(50) null ,  
constraint pk_libro primary key ( id_libro ) );  
  
create table propietario.libro_escritos (  
id_libro      number( 5 ) not null,  
id_autor      number( 5 ) not null,  
constraint pk_libro_escritos primary key ( id_libro, id_autor ) ,  
constraint fk_libro_escrito_libro foreign key ( id_libro ) references propietario.libro (id_libro ) ,  
constraint fk_libro_escrito_autor foreign key ( id_autor ) references propietario.autor (id_autor  
));  
  
create table propietario.biblioteca (  
id_biblioteca number( 5 ) not null,  
nombre        varchar2(50) not null ,  
direccion     varchar2(50) not null ,  
constraint pk_biblioteca primary key ( id_biblioteca ) );  
  
create table propietario.ejemplares (  
id_libro      number( 5 ) not null,  
id_ejemplar   number( 5 ) not null,  
id_biblioteca number( 5 ) not null,  
prestar       char(1) not null ,  
f_compra      date null,  
precio        number(7,2) null,  
notas         varchar2(2048) null,  
constraint pk_ejemplares primary key ( id_libro, id_ejemplar ) ,  
constraint fk_ejemplar_libro foreign key ( id_libro ) references propietario.libro (id_libro ) ,
```



constraint fk_ejemplar_biblioteca foreign key (id_biblioteca) references propietario. biblioteca
(id_biblioteca));

create table propietario.prestar (

id_libro number(5) not null,

id_ejemplar number(5) not null,

nsocio number(5) not null,

f_prestamo date not null,

f_devolucion date null,

constraint pk_prestar primary key (id_libro, id_ejemplar, nsocio, f_prestamo),

constraint fk_prestar_ejemplar foreign key (id_libro, id_ejemplar) references
propietario.ejemplares (id_libro, id_ejemplar),

constraint fk_prestar_socio foreign key (nsocio) references propietario.socio (nsocio));

Respuesta 2 b

CREATE USER usubi1 IDENTIFIED BY passwrod1;

CREATE USER usubi2 IDENTIFIED BY passwrod2;

...

CREATE USER usubiN IDENTIFIED BY passwrodN;

Respuesta 2 c

CREATE ROLE usuario_base ;

CREATE ROLE usuario_gestor ;

CREATE ROLE usuario_administrador ;

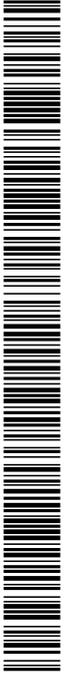
GRANT SELECT, UPDATE on socio TO usuario_base ;

GRANT SELECT, INSERT, UPDATE on socio TO usuario_gestor ;

GRANT SELECT, INSERT, UPDATE, DELETE on socio TO usuario_administrador ;

Código Seguro de Verificación: 6e7b1447-24dc-4533-9adc-
f6730aa77b4d
Origen: Administración
Identificador documento original: ES_L01030149_2025_22087989
Fecha de impresión: 23/04/2025 12:28:56
Página 9 de 9

FIRMAS
1.- DOLORES M ALCARAZ JEREZ (Secretaria del Tribunal), 23/04/2025 12:28



Respuesta 2 d

```
create or replace trigger propietario.trigger_update_socio
    before update on propietario.socio
    for each row
begin
    insert into propietario.hsocio ( nsocio, nombre, ape1, ape2, teléfono, email )
    values( :old. nsocio, :old. nombre, :old. ape1, :old. ape2, :old. teléfono, :old. Email);
end
```