

# Bienvenido a tu Certificación (18/12 - 11 hs)

¡Ya estás un paso más cerca de ser un Analista del Conocimiento - Dimensión Programador!

El examen consta de 5 ejercicios basados en los conocimientos exigidos por el 111 Mil y por la industria, para convertirte en programador junior. Tené en cuenta y leé con mucha atención las siguientes pautas para la correcta realización y aprobación del examen:

- 1- El examen tiene una duración máxima de 2 (dos) horas. Deberás enviar los resultados antes de cumplir ese tiempo.
- 2- Es necesario responder TODOS los ejercicios para poder aprobar el examen.
- 3- El resultado es APROBADO o DESAPROBADO, no tiene puntaje.
- 4- Los veedores estarán presentes para verificar que el examen se tome en las condiciones adecuadas.
- 5- Está prohibido utilizar el chat, el correo electrónico (fuera del uso normal para abrir este formulario), o cualquier página web que no sea este Google Form.
- 6- Tené presente que el teclado no reconoce la tecla "TAB" para escribir código Java, por lo que deberás usar 3 (tres) veces la tecla "ESPACIO" para poner sangrías (indentar/tabular) al alinear el código.
- 7- En caso de no encontrar el tipeo de una tecla, podrás usar el mapa de caracteres, que podés encontrarlo en: "Tecla Windows + R" y en el cuadro de texto escribir "charmap", y apretar "Enter". También podés encontrarlo en "Inicio--> Accesorios --> Herramientas del Sistema --> Mapa de caracteres".
- 8- El examen es individual. No podrás consultar con tus compañeros sobre ningún tema.
- 9- Cuando en el punto siguiente el formulario te pida la dirección de correo electrónico, tenés que declarar el que usaste para inscribirte a la certificación.

En los próximos días recibirás un correo electrónico con los resultados del examen.

Mucha suerte y ¡a trabajar en los ejercicios!

**\*Required**

Email address \*

Your email address

**Anellidos \***

Apellidos

Your answer

Nombres \*

Your answer

DNI (con puntos) \*

Your answer

Sede de Certificación \*

Choose



## Contexto - Venta de Notebooks



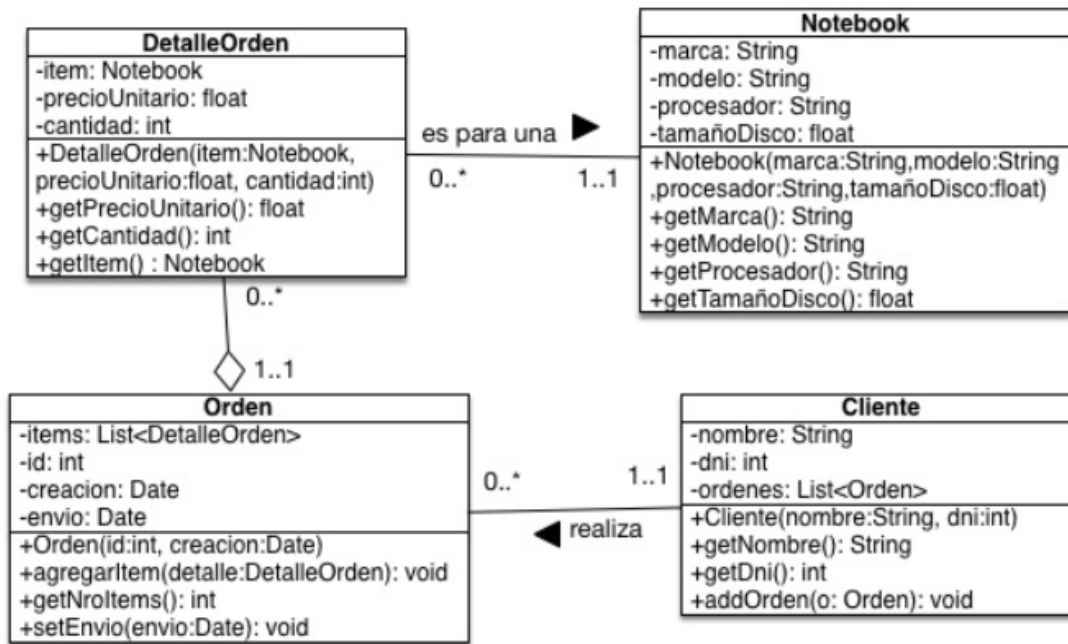
Ana es fanática de la tecnología. Hace dos años decidió abrir su propio negocio de informática llamado "La gran Manzana". El negocio se enfoca principalmente en la venta de notebooks a través de internet. Dada la creciente demanda de notebooks, Ana necesita desarrollar un sistema que la ayude en su negocio.

Como sabe que cursamos el programa 111Mil se puso en contacto con nosotros para que la ayudemos a construir el sistema, que en esta primera etapa del proyecto deberá administrar las órdenes de compra de cada cliente.

## Ejercicio 1 - Implementar desde el diagrama de clases



Dado que el encargado es experto en modelado de software, ha creado un diagrama de clases UML (Lenguaje de Modelado Unificado) para indicarnos lo que se desea. Otros programadores comenzaron con la codificación en Java, pero nos piden que la completemos (solo la clase Notebook está completa).



### 1a- Completar:

```

public class Cliente {
    private List<Orden> ordenes;
    public Cliente(String nombre, int dni) {
        this.nombre=nombre;
        this.dni=dni;
        ordenes=new ArrayList<>();
    }
    public void addOrden(Orden o){
        ordenes.add(o);
    }
}
  
```

**// DEFINIR EL RESTO DE LAS VARIABLES Y CODIFICAR LOS MÉTODOS**

Your answer

### 1b- Completar:

```

public class Orden {
    private List<DetalleOrden> items;
    private Date creacion;
    private Date envio;
    public Orden(int id, Date creacion) {
        this.id=id;
    }
}
  
```

```
        this.creacion=creacion;
        items=new ArrayList<>();
    }
    public int getNroItems(){
        return items.size();
    }
    public void agregarItem(DetalleOrden detalle){
        items.add(detalle);
    }
    // DEFINIR EL RESTO DE LAS VARIABLES Y CODIFICAR LOS MÉTODOS
}
```

Your answer

1c- Completar:

```
public class DetalleOrden {
    // DEFINIR LAS VARIABLES

    public DetalleOrden(Notebook item, float precioUnitario, int cantidad) {
        this.item = item;
        this.precioUnitario = precioUnitario;
        this.cantidad = cantidad;
    }
    // CODIFICAR EL RESTO DE LOS METODOS
}
```

Your answer

```
public class Notebook {
    private String marca;
    private String modelo;
    private String procesador;
    private float tamañoDisco;

    public Notebook(String marca, String modelo, String procesador, float tamañoDisco) {
        this.marca = marca;
    }
}
```

```
this.modelo = modelo;
this.procesador = procesador;
this.tamañoDisco = tamañoDisco;
}
public String getMarca() {
    return marca;
}
public String getModelo() {
    return modelo;
}
public String getProcesador() {
    return procesador;
}
public float getTamañoDisco() {
    return tamañoDisco;
}
}
```

## Ejercicio 2 - Implementar un método a partir de un enunciado

Programar en Java el código del método calcularTotalOrden en la clase Orden. El mismo deberá retornar el precio total de una orden. Para esto deberá tener en cuenta la cantidad solicitada de cada notebook. Por ejemplo, si la orden está compuesta por los siguientes ítems:

- \* 2 notebooks Abble "Pro" cuyo precio unitario es \$20000

- \* 1 notebook Mamung "Z" cuyo precio unitario es \$10000

Entonces, el precio total de la orden será \$50000.

Your answer

## Ejercicio 3 - Interpretación de código

Un desarrollador implementó el siguiente método en la clase Cliente pero no usó nombres representativos.

```
public void xxx() {
    int n = ordenes.size();
    int k;
    for (int m = n; m >= 0; m--) {
        for (int i = 0; i < n - 1; i++) {
            k = i + 1;
            if (ordenes.get(i).getNroItems() > ordenes.get(k).getNroItems()) {
```

```
Orden temp = ordenes.set(i, ordenes.get(k));
ordenes.set(k,temp);
    }
}
}
```

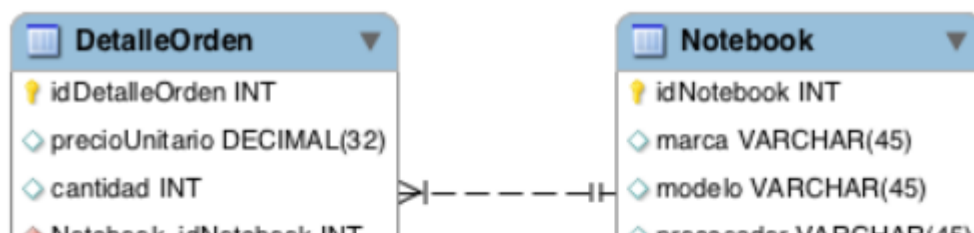
Indique cuál de los siguientes sería el nombre más representativo para el método, dada su funcionalidad:

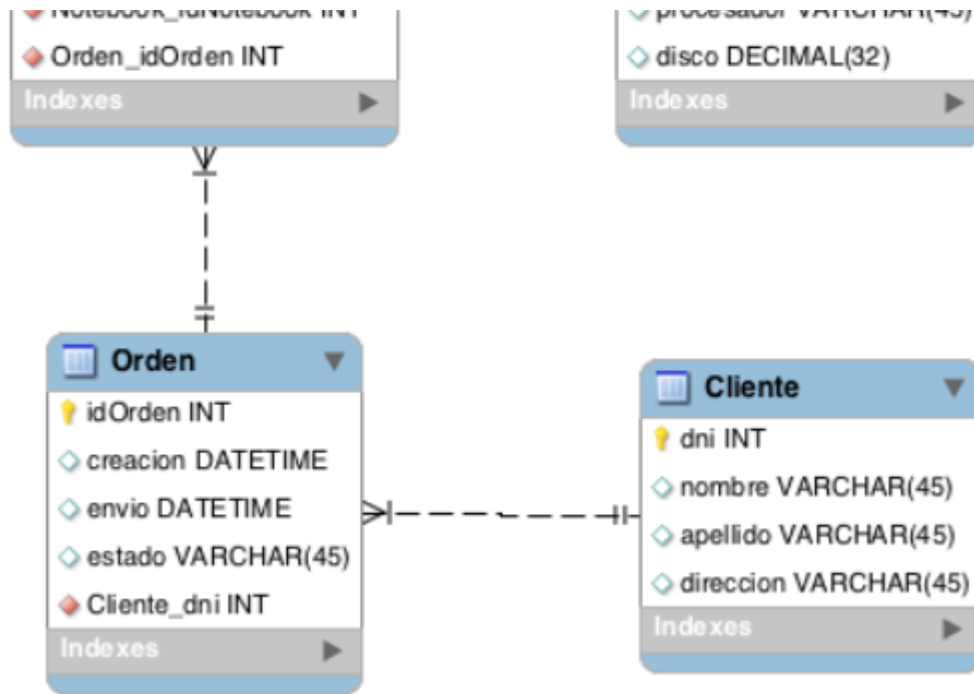
\* Tenga en cuenta que el método `set(i,nuevoElemento)` inserta `nuevoElemento` en la posición `i` de la lista y retorna el elemento que estaba inicialmente en la posición `i`.

- ☐ a. filtrarOrdenes()
- ☐ b. obtenerOrdenConMasItems()
- ☐ c. obtenerOrdenDeMayorPrecio()
- ☐ d. ordenarOrdenesPorCantidadDeItems()
- ☐ e. ordenarOrdenesPorPrecio()

## Ejercicio 4 - Interpretación de Diagrama de Entidades y Relaciones (DER).

A partir del DER, responda Verdadero (V) o Falso (F) a las siguientes afirmaciones (todas las afirmaciones tienen que tener un valor asignado. Puede haber varios V y varios F):





VERDADERO

FALSO

A- Entre las tablas Notebook y DetalleOrden existe una relación 1 a 1

☐☐

B- Entre las tablas Notebook y DetalleOrden existe una relación 1 a N

☐☐

C- Entre las tablas Notebook y DetalleOrden existe una relación N a N

☐☐

D- La clave primaria de la tabla Orden es idOrden

☐☐

E- La clave foránea de la tabla Orden es idOrden

☐☐

F- La tabla Cliente no tiene clave primaria

☐☐

G- La tabla Cliente no tiene clave foránea

☐☐

## Ejercicio 5 - Escribir una consulta SQL.

Dado el diagrama de entidad-relación presentado en el ejercicio anterior, escriba una consulta SQL que liste los números de orden (id) de todas las órdenes pertenecientes a clientes con apellido Rodriguez.

Your answer

Te recordamos revisar tu examen antes de presionar el botón "ENVIAR", ya que una vez que lo hagas, tus respuestas serán enviadas y no podrás modificarlas.



Éxitos!

Page 1 of 1

SUBMIT

Never submit passwords through Google Forms.

This content is neither created nor endorsed by Google. [Report Abuse](#) - [Terms of Service](#) - [Additional Terms](#)

Google Forms



