

Examen Teórico

1. ¿Qué es la arquitectura MVC y cuál es su propósito en el desarrollo de aplicaciones web?

MVC (Modelo-Vista-Controlador) es un patrón arquitectónico que separa la lógica de negocio (Modelo), la presentación (Vista) y la gestión de eventos/flujo (Controlador) para mejorar la mantenibilidad y escalabilidad del código.

2. Explica qué es la programación orientada a objetos (POO) y menciona algunos conceptos clave asociados a ella.

POO es un paradigma que organiza el código en objetos que contienen datos y código. Conceptos clave: encapsulación, herencia, polimorfismo, abstracción, clases y objetos.

3. Menciona algunos patrones de diseño J2EE y explica brevemente su propósito.

Front Controller: gestión centralizada de solicitudes
DAO: abstracción del acceso a datos
Session Facade: simplifica la interfaz de negocio
MVC: separación de responsabilidades

4. ¿Qué son las APIs, sockets y webservices? ¿Cuál es su propósito en el desarrollo de aplicaciones?

Son interfaces para comunicación entre software. Sockets permiten comunicación bidireccional entre procesos. Webservices son APIs que utilizan protocolos web para comunicación entre sistemas distribuidos.

5. ¿Qué significa REST y SOAP en el contexto de servicios web? ¿Cuáles son las principales diferencias entre ellos?

REST usa HTTP con formato ligero (JSON/XML), sin estado, recursos identificados por URLs. SOAP es un protocolo más pesado basado en XML, con estado, más seguro y con contratos WSDL. REST es más simple y flexible, SOAP más estructurado y robusto.

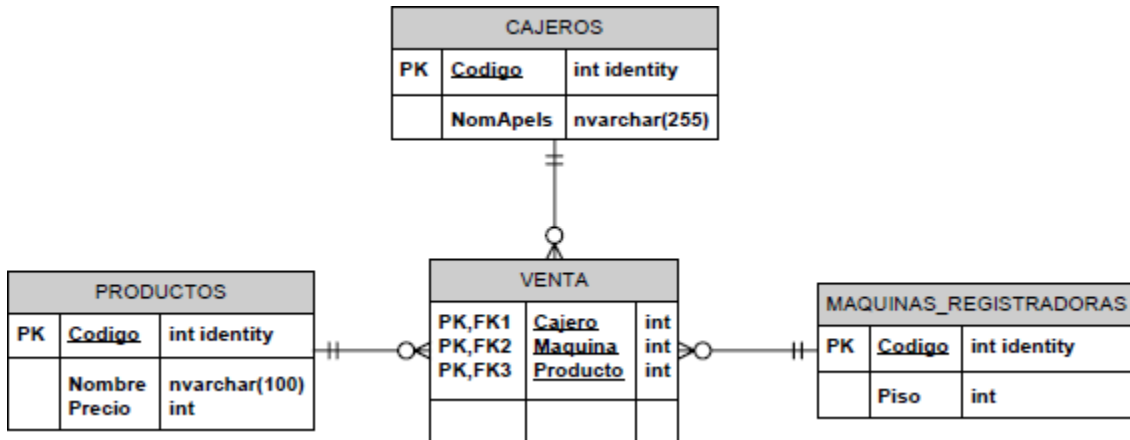
6. ¿Qué es Kubernetes y cuál es su función en el despliegue y gestión de aplicaciones en contenedores?

Es una plataforma open-source para automatizar el despliegue, escalado y gestión de aplicaciones en contenedores. Maneja la orquestación, balanceo de carga, almacenamiento y auto-reparación de contenedores.

Examen Práctico

1. Desarrolla en Java una API REST utilizando el framework Spring Boot que permita realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) sobre una entidad "Usuario" utilizando una base de datos PostgreSQL. Implementando capas de servicios y manejo de excepciones.
2. Desarrolla una función en Node.js que utilice Promesas para leer un archivo CSV, procesar los datos y guardarlos en una base de datos MongoDB.
3. Configurar un archivo de manifiesto de Kubernetes para desplegar una aplicación en un clúster de Kubernetes.

4. Los Grandes Almacenes



Con base en, el ER generar las consultas en PostgreSQL necesarios para solventarlos:

- Generar script para poblar todas las tablas.
- Mostrar el número de ventas de cada producto, ordenado de más a menos ventas.
- Obtener un informe completo de ventas, indicando el nombre del cajero que realizo la venta, nombre y precios de los productos vendidos, y el piso en el que se encuentra la máquina registradora donde se realizó la venta.
- Obtener las ventas totales realizadas en cada piso.
- Obtener el código y nombre de cada cajero junto con el importe total de sus ventas.
- Obtener el código y nombre de aquellos cajeros que hayan realizado ventas en pisos cuyas ventas totales sean inferiores a los 5000 pesos.

Enviar respuestas de examen teórico y codigo fuente de examen práctico.