**Parcial Paradigmas y Lenguajes de programación III**

**Nombre de alumno:** Luciano Steinmann

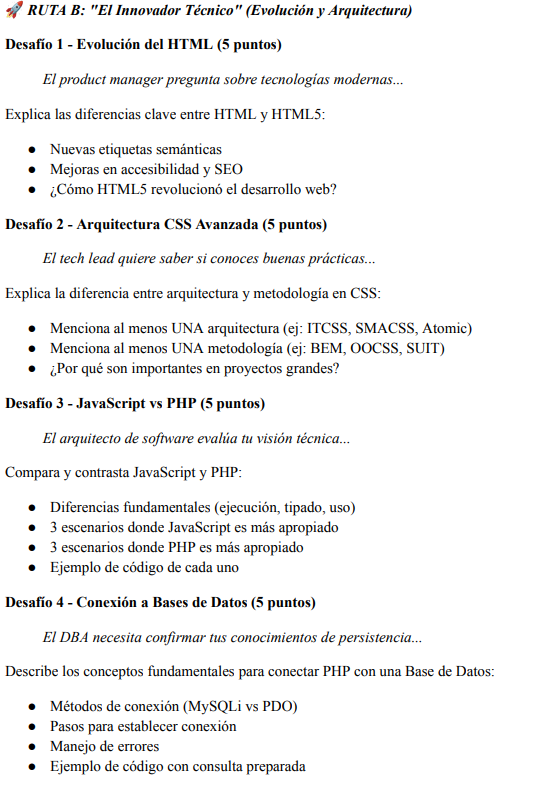
**Profesor:** Mgter. Ing. Agustín Encina

**Cátedra:** Paradigmas y Lenguajes de programación III

**Fecha:** 29/10/25

**Desarrollo**

**Grupo de preguntas elegido:** B



**Respuestas**

1. La diferencia principal es que HTML5 trajo un montón de etiquetas nuevas que dan más significado al contenido, como header, footer, nav y article. Esto ayuda mucho a la accesibilidad porque los lectores de pantalla pueden entender mejor la estructura de la página, y también mejora el SEO porque los buscadores pueden identificar más fácilmente qué parte es el contenido principal. HTML5 realmente cambió todo al agregar capacidades nativas para video y audio, el canvas para dibujar, y APIs para trabajar offline o usar la geolocalización, cosas que antes requerían plugins.

2. La arquitectura es como organizamos los archivos y el código, por ejemplo ITCSS que ordena todo en capas desde lo más general hasta lo específico. La metodología son más las reglas de nomenclatura, como BEM que usa bloques, elementos y modificadores. En proyectos grandes esto es fundamental porque cuando trabajan muchos desarrolladores, si cada uno escribe CSS a su manera se vuelve un caos imposible de mantener.

3. Comparando JavaScript y PHP, son lenguajes bastante diferentes aunque ambos se usan en web. JavaScript corre principalmente en el navegador y se enfoca en la interactividad, mientras que PHP es del lado del servidor y se usa para generar páginas dinámicas. JavaScript es ideal para aplicaciones que necesitan mucha interacción en tiempo real, como chats o interfaces muy dinámicas, mientras que PHP es mejor para cosas como procesar formularios, manejar sesiones de usuarios o conectar con bases de datos. Cada uno tiene su fuerte.

4. Conectar PHP con bases de datos, lo más importante hoy es usar PDO porque es más seguro y funciona con diferentes tipos de bases de datos, no solo MySQL. La clave está en usar consultas preparadas para evitar inyecciones SQL, y manejar bien los errores con try-catch para que no se muestre información sensible al usuario. Básicamente se trata de establecer la conexión con los datos correctos, preparar las consultas con parámetros, ejecutarlas y procesar los resultados, siempre cerrando la conexión después.