Drupal 10 Interview Preparation – Detailed Caching Explanation

Q1: What are the different types of caching available in Drupal 10? Explain in detail.

Drupal 10 provides multiple layers of caching to improve both backend performance and frontend response time.
Below are the major types of caching:

1. Internal Page Cache:
This cache stores fully rendered pages for anonymous users. When enabled, Drupal serves the cached HTML
without bootstrapping the full Drupal framework, significantly improving response time.

2. Dynamic Page Cache:
Unlike Internal Page Cache, Dynamic Page Cache works for both authenticated and anonymous users.
It caches page content while respecting cache contexts such as user roles, permissions, and language.

3. Render Cache:
Render cache stores rendered output of individual render arrays (blocks, views, entities). This avoids rebuilding components repeatedly and improves performance of complex pages.

4. Cache Tags:
Cache tags allow fine■grained invalidation. When content changes (e.g., node update), only related cached
items are cleared instead of the entire cache. This ensures high performance with accurate content updates.

5. Cache Contexts:
Cache contexts define variations of cached content based on conditions like user role, language,
theme, or URL query parameters.

6. Cache Max-Age:
Defines how long content should remain cached. It can be set per render array for granular control.

7. Twig Template Cache:
Drupal compiles Twig templates into PHP and caches them. In production, Twig debug

should be disabled
to maximize template caching performance.

8. External Caching (Reverse Proxy / CDN):
Drupal integrates with Varnish, Nginx reverse proxy, and CDNs to cache full HTTP responses,
reducing server load and improving global performance.

9. Database & Memory Cache Backends:
Drupal supports database caching by default but can be enhanced using Redis or Memcached
for faster in-memory caching.

10. BigPipe:
BigPipe improves perceived performance by sending the page skeleton first and loading dynamic placeholders asynchronously.

In summary, Drupal 10 follows a layered caching architecture combining page-level, component-level,
and backend caching mechanisms to achieve optimal performance, scalability, and user experience.