

Repaso-Primer-Parcial.pdf



Anónimo



Sistemas Distribuidos e Internet



3º Grado en Ingeniería Informática del Software



Escuela de Ingeniería Informática
Universidad de Oviedo

antes



**Descarga sin publi
con 1 coin**



Después

WUOLAH



Importante

Puedo eliminar la publi de este documento con 1 coin

¿Cómo consigo coins? → Plan Turbo: barato
→ Planes pro: más coins

pierdo espacio



Necesito concentración

ali ali ooh
esto con 1 coin me
lo quito yo...

WUOLAH

Repaso SDI
Parcial 1 (2018-2019)

REPASO SDI PARCIAL 1

Temas 1-5 (Hasta Selenium)

Seminarios 1 y 2

¿Qué elemento de una aplicación JEE se encarga de

- Recibir peticiones HTTP
- Redireccionarlas a servlets específicos?

El contenedor de servlets. Se encarga de recibir peticiones y de seleccionar el servlet que se encarga de realizar cada petición.

Nombra dos métodos incluidos en la interfaz Servlet, explica muy brevemente para qué sirven

Se pueden programar `init()` -> `service()` (no recomendado) -> `doGet()` -> `doHead()`

Desde un servlet, explica la diferencia funcional entre obtener un valor "nombre" con:

- a) `request.getParameter("nombre")`: accedes al parámetro independientemente si es post o get
- b) `request.getSession().getParameter("nombre")`: accede al valor del parámetro del usuario que ha iniciado la sesión

Sobre qué elemento básico de JEE se construye una JSP

Un Servlet.

Cuál es la principal ventaja de las JSP sobre tecnologías anteriores

- Mayor separación entre presentación y lógica de negocio.
- Generar HTML por código dificultaba: separación de tareas lógica y diseño.

Qué tres tipos de elementos pueden contener una JSP

- Elementos de secuencia / Scripting -> scriptlets, declaraciones y expresiones.
- Directivas -> cabeceras de las páginas.
- Acciones -> invocaciones de los Beans.

En la siguiente directiva JSP qué es `gestorCanciones`.

```
<jsp:setProperty name="gestorCanciones" property="canciones" value="2"/>
```

Un Bean / el identificador/nombre de un Bean.

```
<jsp:useBean id="gestorCanciones" class="con.gestorCanciones"/>
```

```
<jsp:setProperty name="gestorCanciones" property="canciones" value="2"/>
```

En una arquitectura MVC en JEE qué elemento se utilizará para implementar el modelo (lógica de negocio y datos) Servlets, Beans o JSPs.

Los Beans / Clases Java. En el MVC el modelo son los servicios + repositorios.

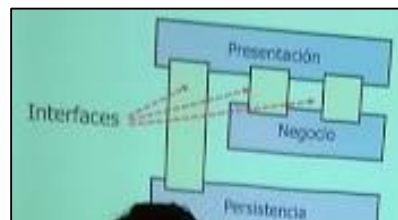
¿En qué consiste el patrón Fachada? ¿Cuál es su principal ventaja?

- Interfaz único simplificado con los servicios más generales del sistema.
- Interfaz simple para un subsistema capa/compleja.
- Reducir el acoplamiento.

¿Con qué patrón se reduce el acoplamiento entre capas?

Patrón Fachada.

Hay algún error en la siguiente aplicación de un modelo n-capas



- Error al pasar de presentación a persistencia. Hay que pasar de manera escalonada.
- Negocio no se comunica con persistencia, aunque podría no hacer falta si no hace falta repositorio.

Completa el siguiente fragmento de código para que la clase Miconfiguración instancie un objeto de la clase calculadora como Bean. Calculadora tiene un constructor sin argumentos:








```
@Configuration
public class MiConfiguracion {
    ...
    @Bean
    ...
}

@Configuration
public class MiConfiguracion {
    @Bean
    public Calculadora calculadora() {
        return new Calculadora();
    }
}
```

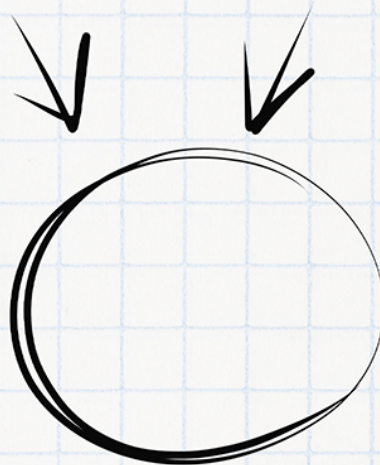
Patrón singleton -> solo hay una instancia de @Service, @Controller, @Repository, @Configuration

Imagínate aprobando el examen

Necesitas tiempo y concentración

Planes	 PLAN TURBO	 PLAN PRO	 PLAN PRO+
 Descargas sin publi al mes	10 	40 	80 
 Elimina el video entre descargas			
 Descarga carpetas			
 Descarga archivos grandes			
 Visualiza apuntes online sin publi			
 Elimina toda la publi web			
 Precios Anual <input type="checkbox"/>	0,99 € / mes	3,99 € / mes	7,99 € / mes

Ahora que puedes conseguirlo,
¿Qué nota vas a sacar?



WUOLAH

¿Quién recibe antes una petición, un controlador o un interceptor?

El interceptor.

¿Cuál es la función del LocaleChangeInterceptor en los sistemas de internacionalización?

Se utiliza para detectar si las peticiones incluyen el parámetro que indica el idioma, ej Lang=es.

Según la siguiente configuración que se requiere para que una petición pueda acceder a la URL /datos/personal?. Razona la respuesta

```
http.  
    .authorizeRequests()  
        .antMatchers("/datos/**").authenticated()  
        .antMatchers("/datos/persona").hasAuthority("ROLE_USUARIO")  
        .antMatchers("/datos/personal").hasAuthority("ROLE_ADMIN")  
        .anyRequest().permitAll()
```

Todos los usuarios autenticados podrán acceder dado que el primer matching es el primero en aparecer (funciona en cascada). Como en la primera línea ponemos **, estamos permitiendo el acceso a cualquier usuario registrado.

```
http.  
    .authorizeRequests()  
        .antMatchers("/barco/add").authenticated()  
        .antMatchers("/barco").hasAuthority("ROLE_USUARIO")  
        .antMatchers("/barco/**").hasAuthority("ROLE_ADMIN")  
        .anyRequest().permitAll()
```

Tener rol de administrador.

¿Qué generará el siguiente fragmento de código Thymeleaf

```
<div sec:authorize="!isAuthenticated()">  
    <p>Bienvenido</p>  
</div>
```

Genera el párrafo bienvenida si no hay ningún usuario autenticado.

¿Qué dos tipos de validaciones de datos de entrada podríamos aplicar? ¿Cuáles son más seguras?

Servidor -> más seguro y cliente -> utiliza código script.

Importante

Puedo eliminar la publi de este documento con 1 coin

¿Cómo consigo coins? → Plan Turbo: barato
→ Planes pro: más coins

pierdo espacio



Necesito concentración

ali ali ooh
esto con 1 coin me
lo quito yo...

WUOLAH

Repaso SDI

Parcial 1 (2018-2019)

Completa el siguiente validador para que se incluya un error en el campo nombre del formulario

```
@Override
public void validate(Object target, Errors errors) {
    User user = _____;
    if (user.getNombre().length() < 5) {
        _____
    }
}

@Override
public void validate(Object target, Errors errors) {
    User user = (User) target;
    if (user.getNombre().length() < 5) {
        errors.rejectValue("nombre", "MENSAJE");
    }
}
```

Qué es el objeto sesión

Un objeto propio de cada cliente/usuario. Se identifica con una id única que viaja en cada petición realizada por el navegador.

Pon un ejemplo de uso común

Una cesta de la compra, aspectos de configuración...

Completa el siguiente código, se debe utilizar HttpSession para guardar la fecha del último inicio de sesión de un usuario

```
@Controller
public MiControlador{
    _____
    @RequestMapping(value="/inicio")
    public String acceder()
    {
        String fecha = new SimpleDateFormat("yyyyMMdd_HHmss")
            .format(Calendar.getInstance().getTime());
        _____
        return "inicio";
    }
}

@Controller
public MiControlador
{
    @Autowired
    private HttpSession httpSession;

    @RequestMapping(value="/inicio")
    public String acceder()
    {
        String fecha = new SimpleDateFormat("yyyyMMdd_HHmss")
            .format(Calendar.getInstance().getTime());
        httpSession.setAttribute("clave", fecha);
        return "inicio";
    }
}
```

¿Como afectaría a los formularios de la aplicación incluir protección contra ataques CSRF?

Debeían incluir un parámetro nuevo con el token CSRF. EL token es generado por la aplicación.

Crea una versión equivalente del método findAllByUsuario (implementado en un repositorio) para que pasea utilizar un sistema de paginación.

```
List<Coche> findAllByUsuario (Usuario usuario );  
  
Page<Coche> findAllByUsuario (Pageable pageable, Usuario usuario);
```

El siguiente servicio debería permitir que un usuario vea los detalles de una de sus propias asignaturas, ¿hay algún aspecto de seguridad que pueda ser mejorado?

```
@Service  
public MiServicio {  
    public Asignatura getMiAsignatura (Long id){  
        Asignatura asignatura =  
            asignaturasRepository.findOne(id);  
        return asignatura;  
    }  
}
```