

# DCC156 Laboratório IV

Depto. Ciência da Computação

[www.ufjf.br/ciro\\_barbosa](http://www.ufjf.br/ciro_barbosa)

# Java EE

Consulte a referência [1] para responder as questões abaixo

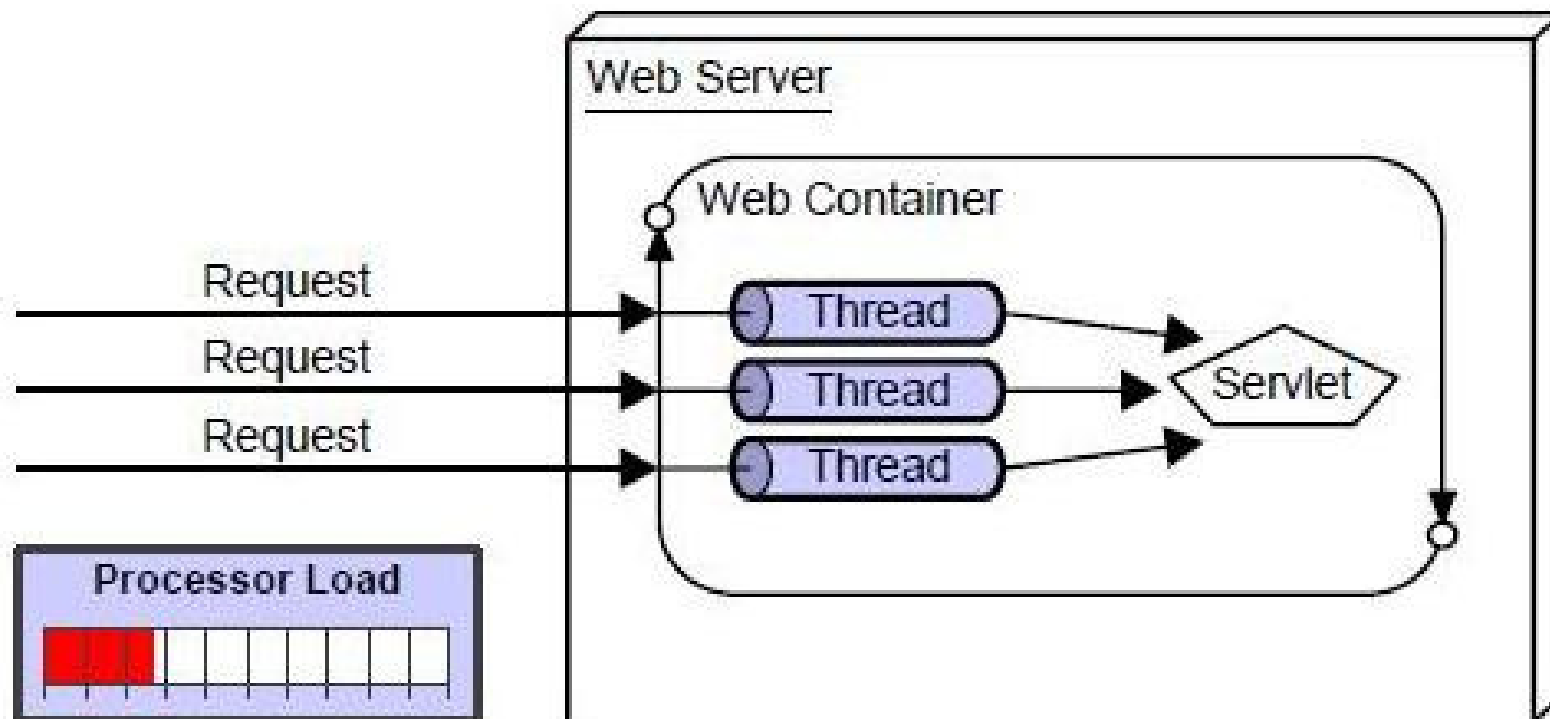
1. O que é o Java Enterprise Edition?
2. Diferenciar um Servidor de Aplicação de um Servlet Container.
3. Cite 4 especificações do Java EE
4. Cite 2 exemplos de servidores de aplicação.
5. Cite 2 exemplos de Servlet Container.

# Servlets

6. Como capturar exceções do servlet através de recursos do web.xml?
7. Como capturar erros HTTP em sua aplicação web?
8. Quais as vantagens de se usar Servlet em comparação com GCI?
9. Explique porque é preciso haver controle de concorrência no uso de Servlets.

Enviar respostas para [ciro.barbosa@ice.ufjf.br](mailto:ciro.barbosa@ice.ufjf.br), com subject “exercício lp4”

# Advantage of Servlet



# ServletConfig Interface

- An object of ServletConfig is created by the web container for each servlet. This object can be used to get configuration information from web.xml file.

```
ServletConfig config=getServletConfig();
```

# ServletContext Interface

- An object of ServletContext is created by the web container at time of deploying the project. This object can be used to get configuration information from web.xml file. There is only one ServletContext object per web application.

//creating ServletContext object

```
ServletContext context=getServletContext();
```

# Parâmetros init do Servlet

Você pode passar parâmetros para um servlet do web.xml file. Os parâmetros init de um servlet podem apenas ser acessados por aquele servlet. Aqui está como você os configura no arquivo web.xml:

```
<servlet>
  <servlet-name>controlServlet</servlet-name>
  <servlet-class>butterfly.ControlServlet</servlet-class>
  <init-param>
    <param-name>myParam</param-name>
    <param-value>paramValue</param-value>
  </init-param>
</servlet>
```

# Servlet Init Parameters

Aqui está como ler os parâmetros init dentro do servlet, no método `init()`. Esses parâmetros ficam disponíveis para um servlet específico.

```
public class SimpleServlet extends HttpServlet {  
    protected String myParam = null;  
    @Override  
    public void init(ServletConfig config) throws ServletException {  
        super.init(config);  
        this.myParam = config.getInitParameter("myParam");  
    }  
    ...  
}
```



# Context Parameters

Você pode também definir *context parameters*, que estarão acessíveis para toda a aplicação web.

```
<context-param>  
  <param-name>myParam</param-name>  
  <param-value>the value</param-value>  
</context-param>
```

# Context Parameters

Aqui está como você acessa esses parâmetros:

```
String myContextParam = request.getSession().getServletContext()  
    .getInitParameter("myParam");
```

# Redirecionando solicitações

```
DoGet(HttpServletRequest request,  
        HttpServletResponse response){  
    ...  
    response .sendRedirect("http://www.ufjf.br");  
    ...  
}
```

# Arquivos *welcome*

É possível especificar uma lista ordenada de arquivos *welcome* a ser carregado quando o URL de solicitação não é mapeado para um *servlet*:

```
<welcome-file-list>  
    <welcome-file>index.html</welcome-file>  
    <welcome-file>index.htm</welcome-file>  
</welcome-file-list>
```

# Persistência de Dados

Usaremos inicialmente o Banco de Dados MySQL para realização de persistência de dados.

- Seguir tutorial de instalação do MySQL [2]
- Veja tutorial do MySQL Workbench [3]
- Opcionalmente, instalar XAMPP, desabilitando o seu web-server. [4]
- Acessando MySQL no Neatbeans [6]

MySQL Workbench

Local instance MySQL56 x

File Edit View Query Database Server Tools Scripting Help

Navigator: Query 1 Administration - Server Status x

**MANAGEMENT**

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

**INSTANCE**

- Startup / Shutdown
- Server Logs
- Options File

**PERFORMANCE**

- Dashboard
- Performance Reports
- Performance Schema Setup

**SCHEMAS**

Filter objects

- information\_schema
- performance\_schema
- sakila
- test
- world

Information: No object selected

Object Info Session

Connection Name: Local instance MySQL56

Host: cbbarbosa  
 Socket: MySQL  
 Port: 3306  
 Version: 5.6.17  
 MySQL Community Server (GPL)  
 Compiled For: Win64 (x86\_64)  
 Running Since: Sun Apr 06 20:29:41 2014 (0:01)

Refresh

**Available Server Features**

Performance Schema:	<input checked="" type="radio"/> On	SSL Availability:	<input type="radio"/> Off
Thread Pool:	<input type="radio"/> n/a	Windows Authentication:	<input type="radio"/> Off
Memcached Plugin:	<input type="radio"/> n/a	Password Validation:	<input type="radio"/> n/a
Semisync Replication Plugin:	<input type="radio"/> n/a	Audit Log:	<input type="radio"/> n/a

**Server Directories**

Base Directory: C:\Program Files\MySQL\MySQL Server 5.6\  
 Data Directory: C:\ProgramData\MySQL\MySQL Server 5.6\data\  
 Disk Space in Data Dir: 392.00 GB of 441.00 GB available  
 Plugins Directory: C:\Program Files\MySQL\MySQL Server 5.6\lib\plugin\  
 Tmp Directory: C:\Windows\SERVIC~2\NETWOR~1\AppData\Local\Temp\  
 Error Log: ☒ On .\CBBARBOSA.err  
 General Log: ☐ Off

Server Status: Running  
 CPU: 9%  
 Connections: 4  
 Traffic: 9.09 KB/s  
 Key Efficiency: 100.0%  
 Queries per Second: 0  
 InnoDB Buffer Usage: 24.3%  
 InnoDB Reads per Second: 0  
 InnoDB Writes per Second: 0

# Referências

- [1] Caelum. <http://www.caelum.com.br/apostila-java-web/>
- [2] Tutorial instalação MySQL.  
<http://www.devmedia.com.br/instalando-e-configurando-a-nova-versao-do-mysql/25813>
- [3] Workbench online manual.  
<http://dev.mysql.com/doc/workbench/en/index.html>
- [4] XAMPP. [https://www.apachefriends.org/pt\\_br/index.html](https://www.apachefriends.org/pt_br/index.html)
- [5] JavaTpoint. <http://www.javatpoint.com/servlet-tutorial>
- [6] Netbeans. [https://netbeans.org/kb/docs/ide/mysql\\_pt\\_BR.html](https://netbeans.org/kb/docs/ide/mysql_pt_BR.html)