CEGEP VANIER COLLEGE
CENTRE FOR CONTINUING EDUCATION
Web Services
420-941-VA

Teacher: Samir Chebbine **Lab 1** **Sep 20, 2024**

Lab 1: Creating Dynamic Web Project in Eclipse using Java Jakarta EE Platform

Create and Submit a Word file ***Lab1OOPWebServicesYourName.doc*** which contains Answers of theory questions if any and output screenshots for every Java EE Project. Submit the Java projects too and submit the whole Lab 1 as compressed zip file.

1. Creating Servlet in Dynamic Web Project: WebHelloWorldProject

a) Create a new Dynamic Web project called **WebHelloWorldProject**. Create new package called **webHelloWorld** as shown in Figure 1. Notice the Servlet mapping **(/WebHello)** that will be use as URL mapping.

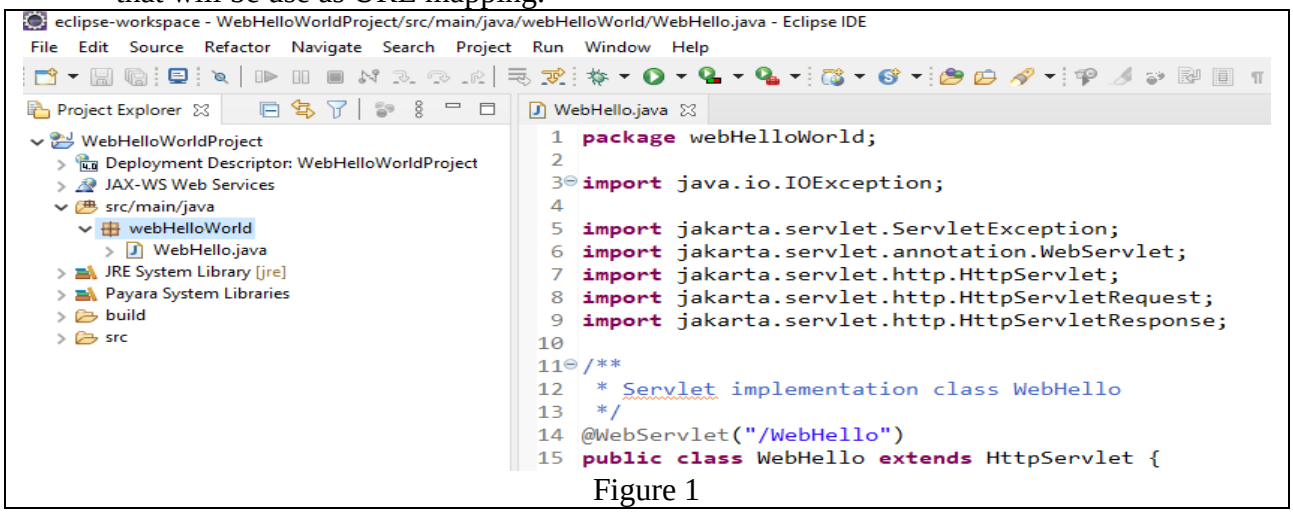


Figure 1

b) Create a new servlet class **WebHello.** Add appropriate statements to print a message output on web browser as shown in Figure 2.

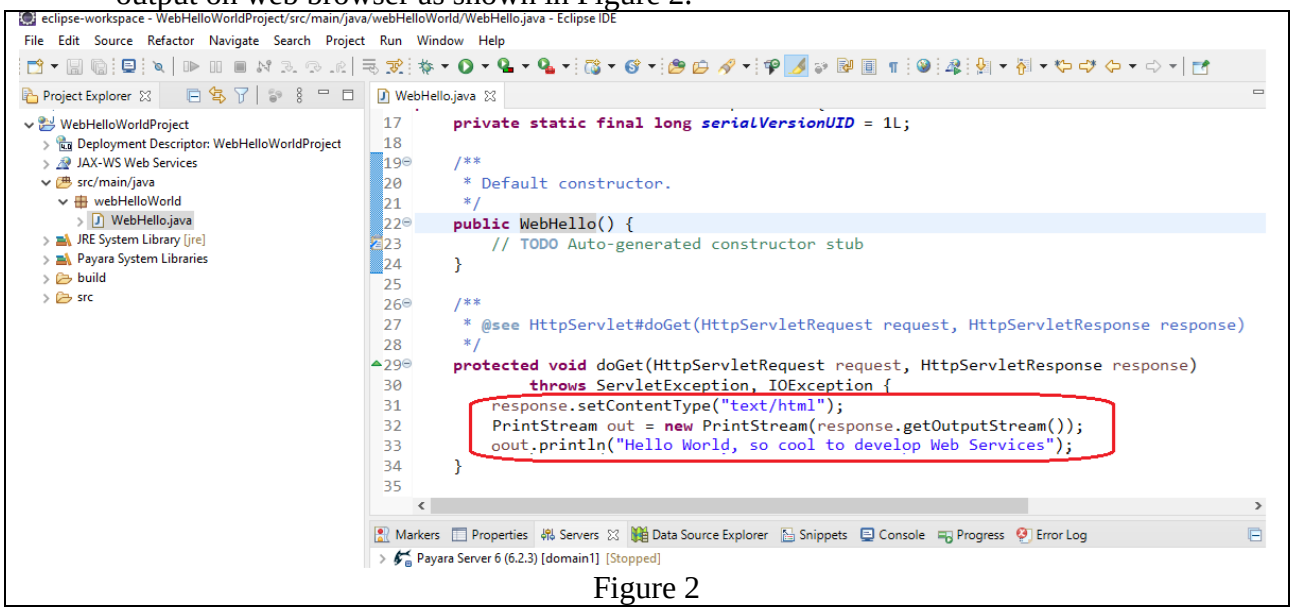


Figure 2

c) Deploy Dynamic Web Project into GlassFish Server and check from browser Console GlassFish Administration that **WebHelloWorldProject** is deployed as shown in Figure 3.

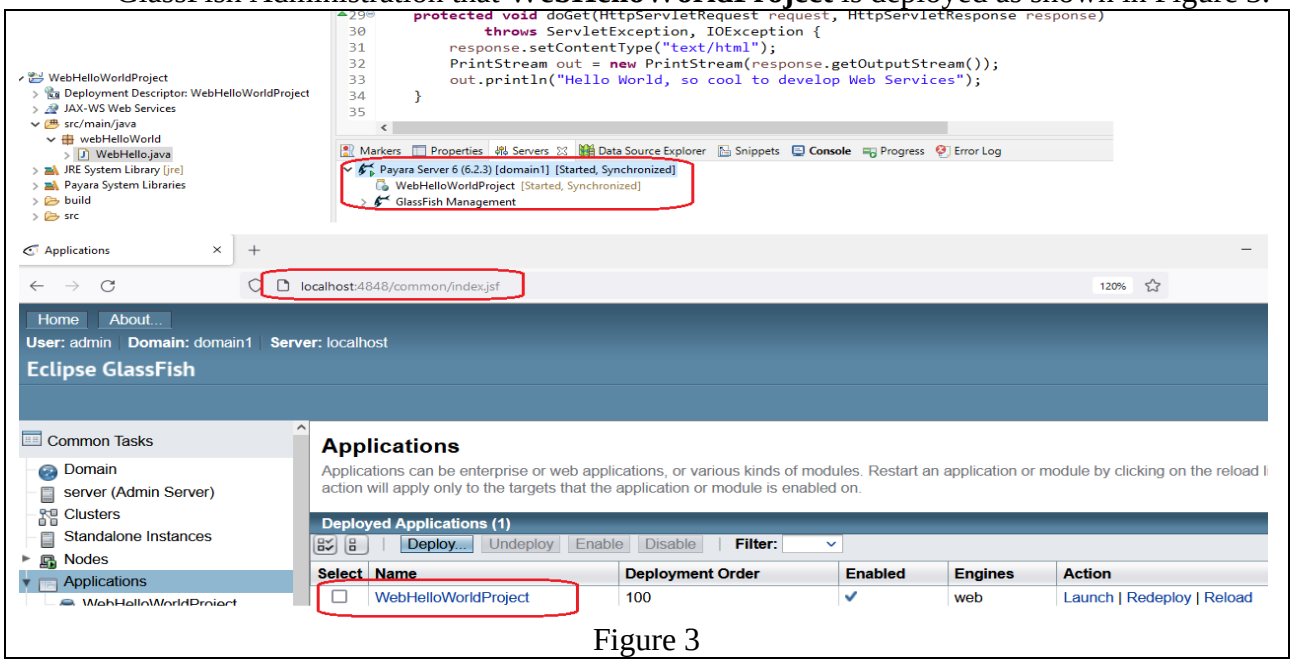


Figure 3

d) Check the deployed Web App **WebHelloWorldProject from browser by sending HTTP GET message** as shown in Figure 4.

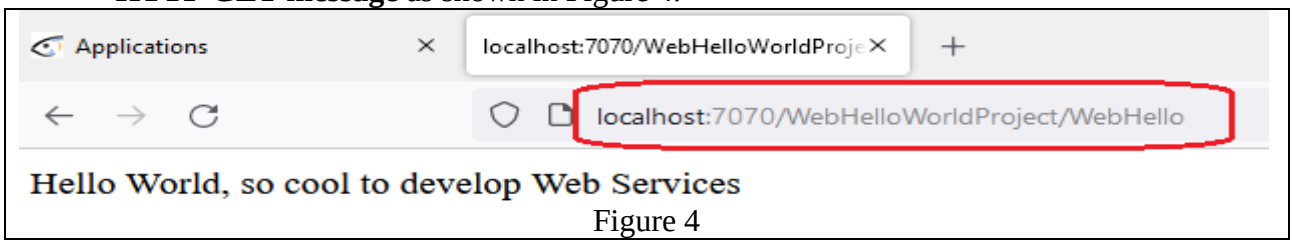


Figure 4

e) Add HTML code within the same Servlet program and instantiate an object from University class to generate the following output as shown in Figure 5.

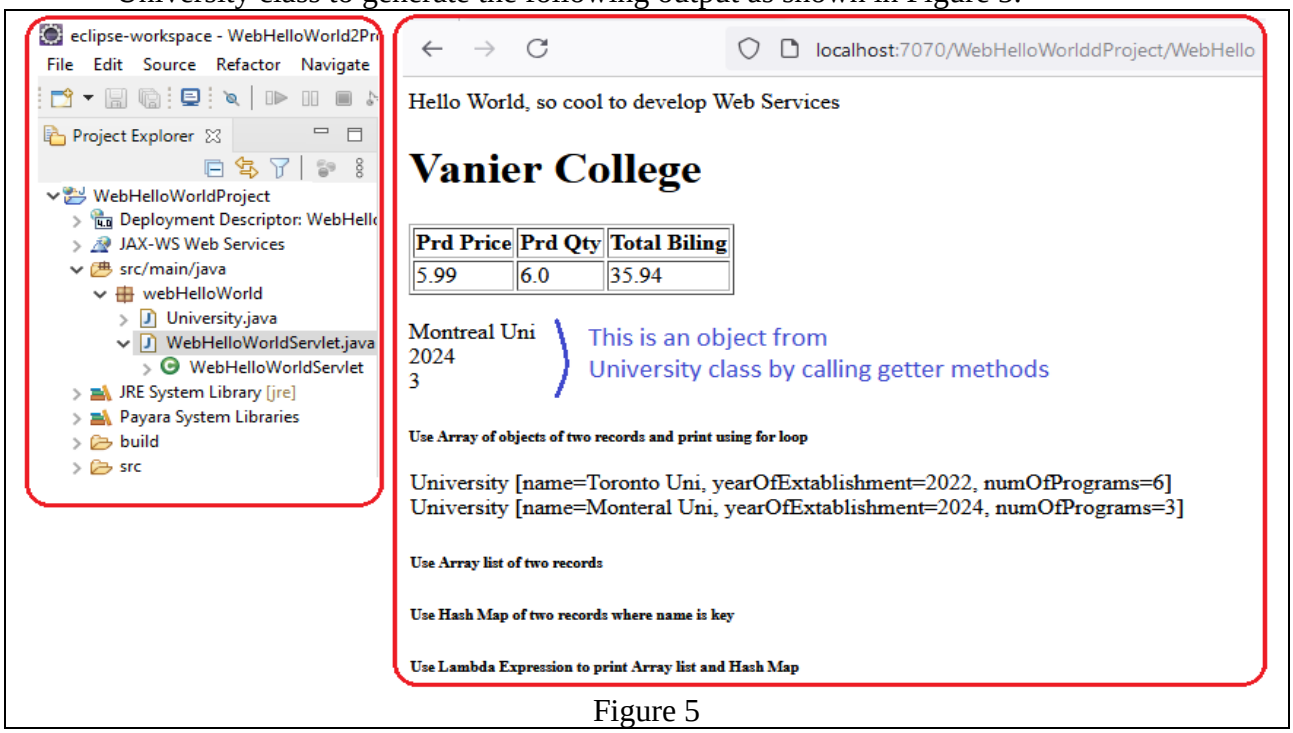


Figure 5

2. Dynamic Web Project: WebBillingProject

a) Create a new Dynamic Web project called **WebBillingProject**. Create new package called **webBilling** as shown hereafter.

b) You need to develop a **Java class** called `Billing`, which takes `client_LName, and client_FName, product_Name, prd_Price=0, prd_Qty` as **private** `non static` members. The variables called `Fed_Tax`, `Prv_Tax` as **public** and `static` data members. The `Billing` class contains the following method members:

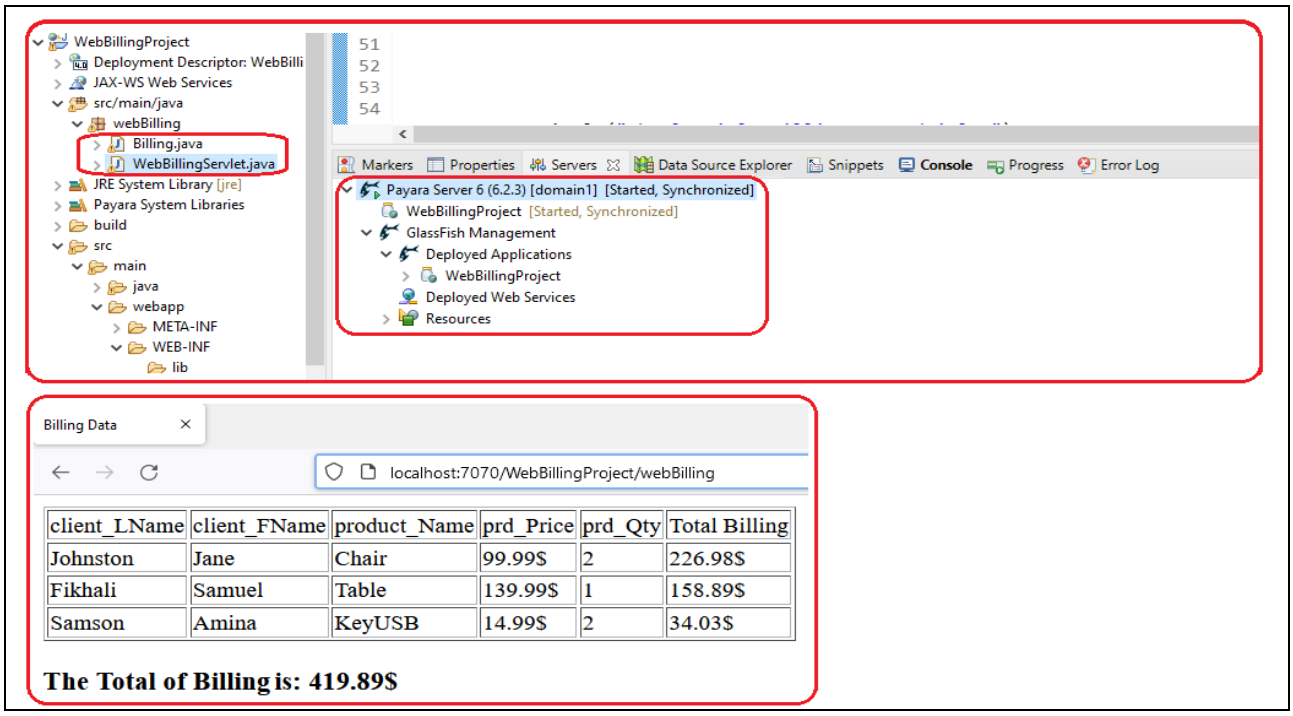
- Add default constructor `(client_LName, client_FName, product_Name prd_Price=0, prd_Qty=0)` and constructor with parameters within the Billing class in order to initialize the data members of every object.
- Add public setter methods and getter methods `(setClient_LName()…,getClient_LName()…,)` in Billing class to modify the values of private members.
- Add a method called `CalculateBilling()` in Billing class to calculate the total of billing

```
T_Billing = (prd_Price* prd_Qty) + (prd_Price*prd_Qty)* Fed_Tax + (prd_Price*prd_Qty)* Prv_Tax
```

c) Create a new servlet class **WebBillingServlet.java**, add appropriate statements to instantiate an `array of object of Billing class type` with three components to be referenced by (`BillingRecords`) using the default constructor. Set every component using the implemented setter methods (`setClient_LName(),setClient_FName(),setproduct_Name(), setPrd_Price(), setPrd_Qty()`) to the following values:

```
Component at index 0: client_LName=Johnston, client_FName=Jane,  product_Name=Chair
       prd_Price = 99.99    prd_Qty=2 Fed_Tax=0.075     Prv_Tax= 0.06
Component at index 1: client_LName=Fikhali, client_FName=Samuel, product_Name=Table
       prd_Price = 139.99   prd_Qty=1 Fed_Tax=0.075     Prv_Tax= 0.06
Component at index 2: client_LName= Samson, client_FName= Amina, product_Name=
KeyUSB prd_Price = 14.99    prd_Qty=2 Fed_Tax=0.075     Prv_Tax= 0.06
```

- Skip through array object (`BillingRecords`) and display its components into Web Table in a browser as shown hereafter.



3. Dynamic Web Project: WebCourseProject

a) Create a new Dynamic Web project called **WebCourseProject**. Create new package called **webCourse** as shown hereafter.

b) You need to develop a **Java class** called `Course` that *represents* a template of the fields used in defining the columns of a given table *Course* which takes `course_no, course_name, max_enrl` as **private** `non static` data members. `credits` as **public** and `static` data member. The `Course` class contains the following method members:

- Add **constructor with parameters** within the `Course` class to initialize the **private** data members `(course_no, course_name, max_enrl, credits)` of every object.
- Add public **Mutator** (**setter**) methods in `Course` class to modify the values of private members.
- Add public **Accessor** (**getter**) methods in `Course` class to read the values of private members.
- Add a return method called `CalculateTotalFees()`in `Course` class to return the total fees of all enrolled students according to the following formula `max_enrl*250`.

c) Create a new servlet class **WebCourseServlet.java**, add appropriate statements to instantiate an `array of object of Course class type` with seven components to be referenced by (`CourseRecords`) using the default constructor. Set every component using the implemented setter methods to the following values shown in Figure hereafter.

- Skip through array object (`CourseRecords`) and display its components into Web Table in a browser as shown hereafter.