James Hopkins

A00220384

Distributed Sytems Project

Lecturer: Robert Nolan

Table of Contents

[Introduction. 1](#_Toc512647847)

[Functionality. 1](#_Toc512647848)

# Introduction.

**Analysis of brief.**

To create a restful java service with a client gui that contains;

* A database with multiple tables and/or tables created using code at runtime plus code to select, insert, update and delete data/
* A JAX-RS/Jersey client that sends all of the HTTP requests GET/PUT/POST/DELETE, parses the response using XMLPullParser and outputs to the GUI.
* A tomcat server that responds to all of the HTTP requests GET/PUT/POST/DELETE.
* GUI that handles the GET/PUT/POST/DELETE actions.

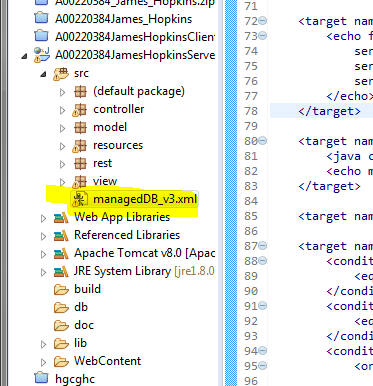
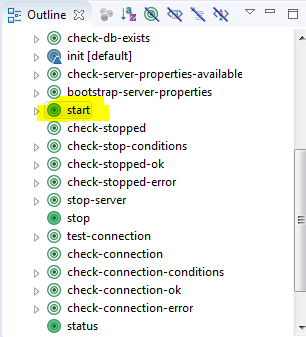
# Functionality.

**Introduction**

This chapter covers the functionality within my project and how to test this functionality.

**Set up**

Right click open with ant editor and right click start run as ant build.

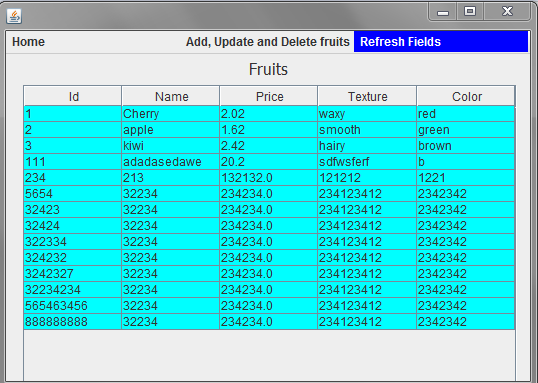
 

Run the application on tomcat.

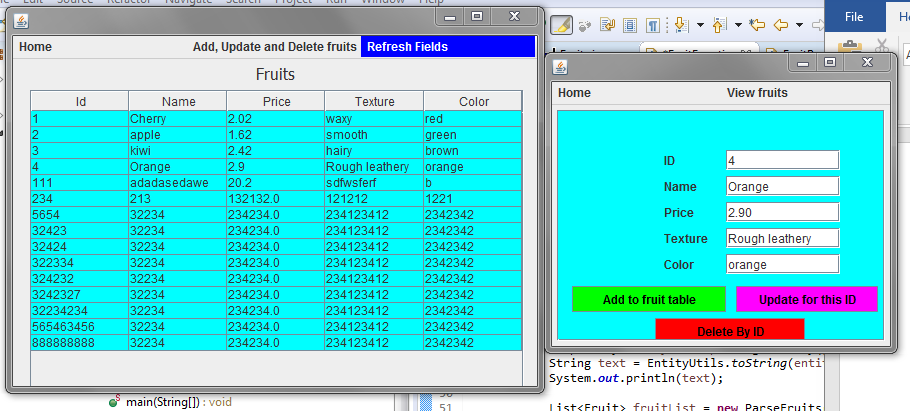
Go to vie.fruits.java run as java application.

**Functionality according to brief.**

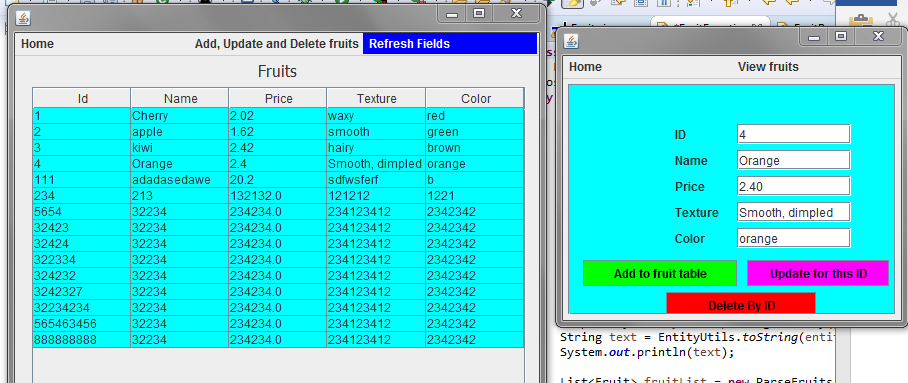
“GUI that handles the GET”



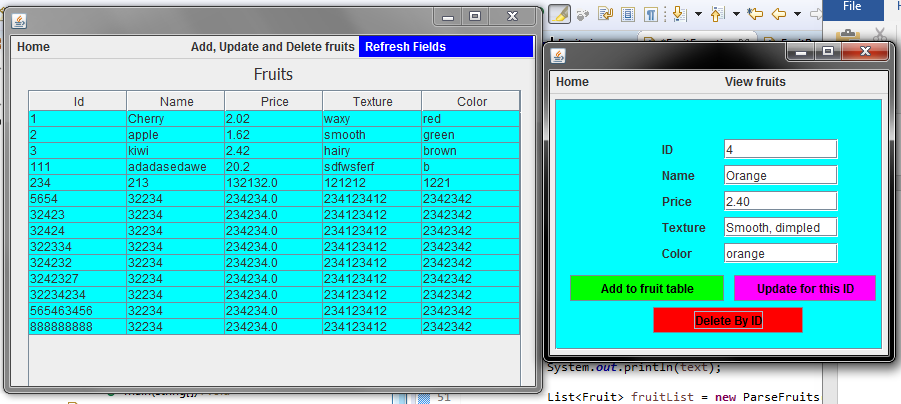
”/POST”

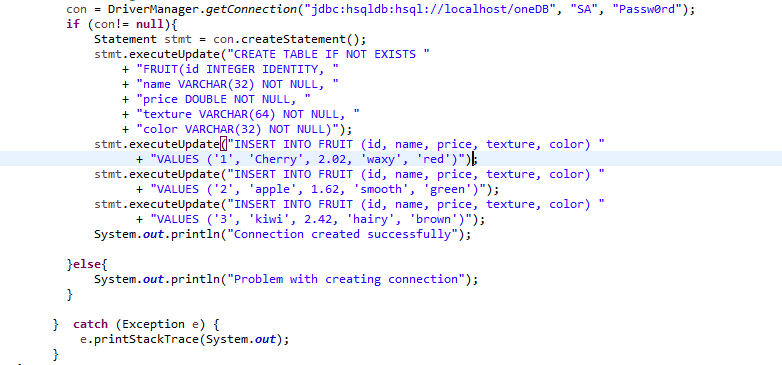


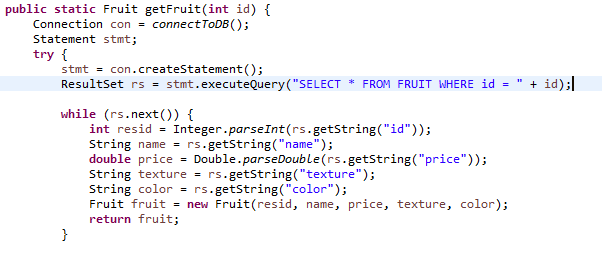
“/PUT”

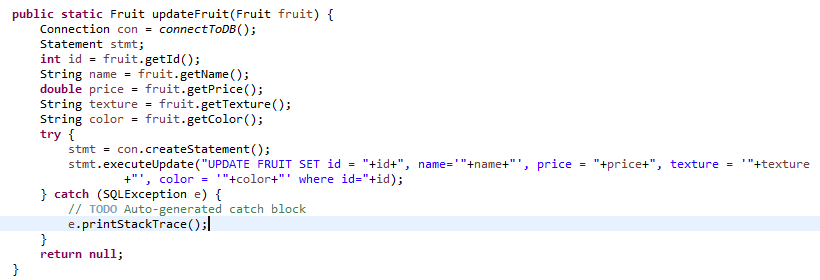


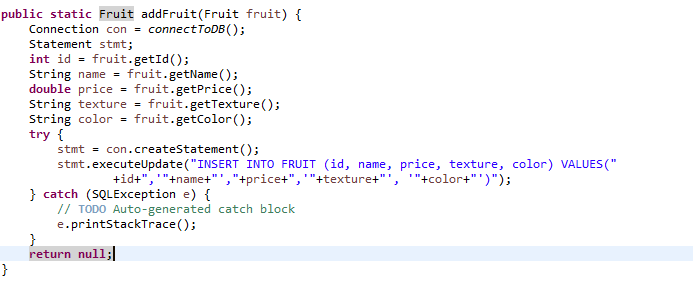
”/DELETE”

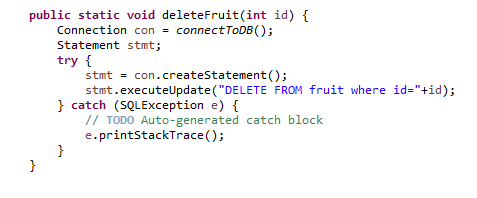


“Multiple tables and/or tables created using code at runtime” 

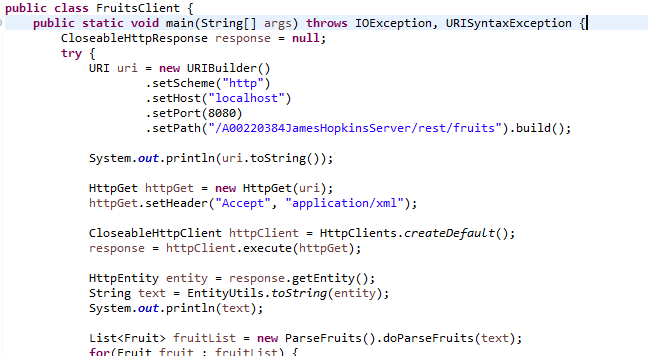
“Plus code to select” 

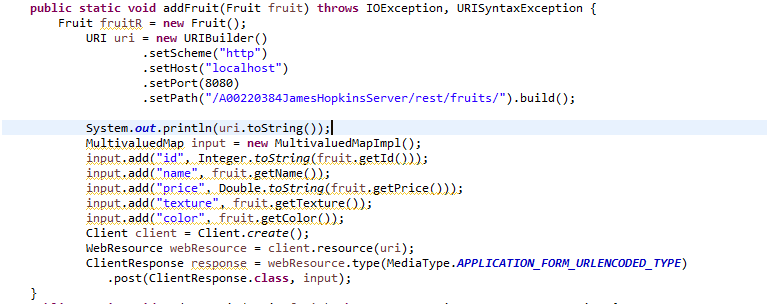
“Update” 

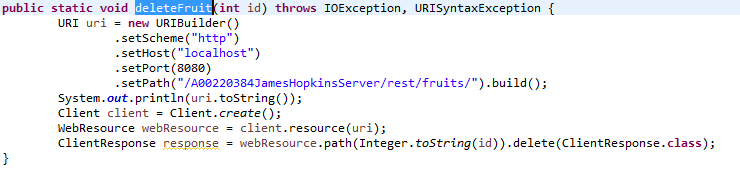
“Insert” 

“Delete”

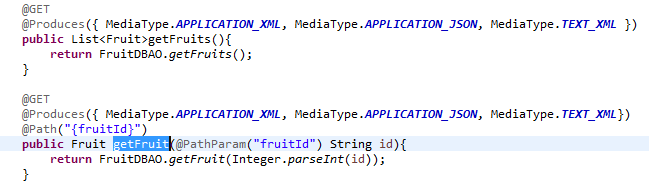
“A JAX-RS/Jersey client that sends all of the HTTP requests GET/”

“PUT” 

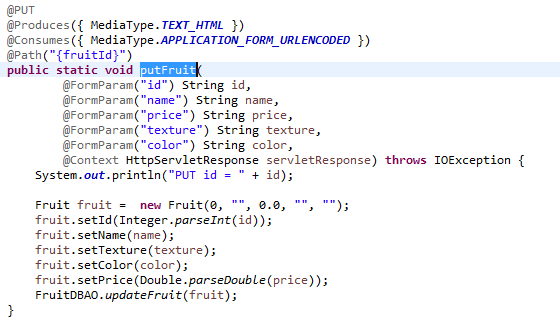
“POST” 

“DELETE” 

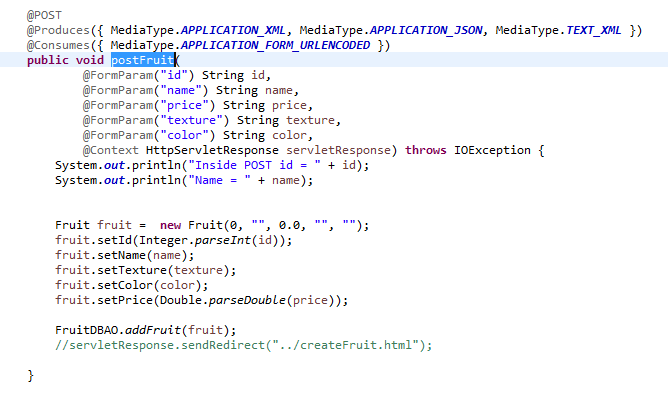
“A tomcat server that responds to all of the HTTP requests, GET”



“PUT”



“POST”



“DELETE” 