IS217 Final Exam

Kazuru Takase

Answer the following questions in 1-2 paragraphs. Each one is worth 5 points.

1. What is a software design pattern? Why are they important?
   1. Design pattern are reusable template used for solving common problem in programming. That way, we can apply the design to our own projects.
   2. It is important because it will easier to find the problem in the codes. Using design pattern will also provide us the common vocabulary necessary to communicate with other programmers.
2. What is unit testing? Why is it important? How would you use it?
   1. Unit testing is a testing of an individual function. It is a method by which individual units of source code are tested to see whether they are fit for use.
   2. It is important because using this will allow you to make big changes fast. Unit testing will also will give me a feedback, and make it easier to pick up where you left off after the code breaks.
   3. I would use it by creating a set of inputs and the expected outputs. From there, I will gradually tests which will handle common cases that are not covered yet. I will also test for failure conditions.
3. Describe the relationship between HTML, CSS, and JavaScript.
   1. The HTML is the layout of the web page. CSS is what makes the webpage visually appealing. In the HTML, Java script manipulated the Document Object Models in the document.
4. Describe the purpose of the Singleton design pattern.
   1. The purpose of the Singleton pattern is to restrict instances of a class to a single object. This is useful when only a single object is needed to maintain certain functionality available throughout a complex system. Singleton also serves as a namespace and provides a single point of access for functionalities.
5. Describe the purpose of the Factory design pattern.
   1. The purpose of a Factory pattern is to provide a generic interface for creating objects. This is useful when an object or component setup is very complex, and needs to generate different copy of objects easily, depending on the environment.
6. Describe the purpose of the publish and subscribe pattern.
   1. The purpose of the Pub/Sub pattern is to reduce coupling between the codes. It is a messaging pattern where the publisher does not program the message to the subscriber, or the receiver. The published messages are characterized into different classes without knowing whether there are people who are subscribed. The pattern will notify to the subscriber if there are any updates in the message from the publisher.
7. Describe the purpose of the decorator pattern.
   1. The purpose of decorator pattern is to promote the reuse of codes. It adds behaviors to existing classes in a system dynamically. Decorator also modifies existing systems where we wish to add additional features to objects without heavily modifying the underlying code using them
8. Write the JavaScript code that illustrates a decorator pattern.
   1. var Car = function() {
   2. console.log('Assemble');
   3. };
   4. Car.prototype = {
   5. start: function() {
   6. console.log('Engine Start')
   7. }
   8. var CarDecorator.prototype = function(car){
   9. this.car = car;
   10. };
   11. CarDecorator.prototype = {
   12. start: fucntion() {
   13. this.car.start();
   14. };
   15. var LockDecorator = function(car) {
   16. CarDecorator.call(this.car);
   17. console.log('Assemble Parts: Lock');
   18. };
   19. LockDecorator.prototype = new CarDecorator();
   20. var WindowDecorator= function(car) {
   21. CarDecorator.call(this.car);
   22. console.log('Assemble Parts: Window');
   23. };
   24. WindowDecorator.prototype = new CarDecorator();
   25. var car = new Car();
   26. car = new WindowDecorator();
   27. car = new LockDecorator();
   28. car.start();
9. Write the JavaScript code that illustrates a factory pattern.
   1. var carFactory = {
   2. makeCar: function(feature) {
   3. var car = new Car();
   4. if (feature && feature.length) {
   5. var i =0,
   6. l = feature.length;
   8. for (i; i< 1; i++) {
   9. var feature = features[i];
   10. switch(features) {
   11. case 'window':
   12. car.new Window(car);
   13. break;
   14. case 'lock':
   15. car.new Lock(car);
   16. break;
   17. }
   18. }
   19. }
   20. return car;
   21. }
   22. }
10. Write JavaScript pseudo code that illustrates the singleton design pattern.
    1. class MySingleton
    2. {
    3. In public:
    4. static MySingleton \* GetInstance() //static method that returns only instance of MySingletone
    5. {
    6. if (OneInstance == NULL) // if not yet instantiated
    7. {
    8. OneInstance = new MySingleton(); //create one and only object
    9. }
    10. return OneInstance;
    11. }
    12. In private:
    13. static MySingleton \* OneInstance; //holds one and only object of MySingleton
    14. MySingleton(); // private constructor
    15. In public: // MySingleton functionalities
    16. void hello();
    17. bool good\_bye();
    18. int see\_you\_later();
    19. };
11. What is jQuery and provide examples of why you would use it? When would you not choose to you it?
    1. jQuery is a free, open source software. It is used to make navigation easier in a document, select DOM elements, standardize AJAX, and even create animations. It is also cross browser compatible.
    2. I would use it for event handling and standardize AJAX.
    3. I would not use jQuery when in larger project, where jQuery is not structured and the codes will be disorganized.
12. What is Backbone.js and how is it different than jQuery
    1. Backbone.js is a template that represents the Model View Collections. It uses the routers as the controller.
    2. Backbone is different from jQuery in that jQuery is a DOM manipulation library. It is also not a programming language. Backbone, on the other hand, allows you to organize and structure the code into data.
13. Write the JavaScript code to select an element by tag.
    1. Var apple = document.getElementByTagName(“tag\_apple”);
14. Write the JavaScript code to select by ID
    1. Var book = document.getElementById(“element\_book”);
15. Write the JavaScript code to select an id and then add html to it.
    1. <html>

…

<script>

Function changeText() {

Document.getElementById(‘element\_text’).innerHTML = ‘Hello’;

}

</script>

<p>How do you say <b id=’element\_text’> Good Bye /b></p>

<input type=’button’ onclick=’chingeText()’ value=’Change Text’/>

1. Write the JavaScript code to create an element.
   1. Var list\_fruit = document.createElement(“List”);
   2. Var link\_pear= document.createElement(“a”);
2. What is Node.js?
   1. Node.js is a platform for building fast, and scalable networks easily. It uses event-driven IO model for lightweight and efficient real-time application that runs across distributed device.
3. What is the difference between unit and functional testing?
   1. Unit testing is testing individual functions. Functional testing is testing part of the program for their functionality. In functional testing, if there’s an action, there will be a response

Answer the following questions in 2-3 paragraphs. Each one is worth 10 points.

1. You have been hired to design and manage a team of developers tasked with creating a web application. How would you explain to your developer the importance of using standard design patterns when designing the system? Provide some practical examples that illustrate to your team how you will use the concept of design pattern within the project.
   1. Design pattern is a template that everyone can use. Finding the mistakes will be easier. Using design patter will keep your work organized and easy to read.
   2. There will always be an error in a program. If I use a design pattern, I can look at other people’s work and the approaches. I can also discuss with other programmer with the problem that I’m having. We will share common vocabulary, so explaining the problem and the possible solution will be easier.
2. You have been hired to design and manage a team of developers tasked with creating a web application. How would you explain to your developer the importance of creating unit tests? Provide some practical examples that illustrate to your team why unit testing is important.
   1. Unit testing will be a way to test modules without the application. It is a way to make sure that each change in the code will not impact the overall program negatively. Using it will also make you understand the outline of the code, which is important if you’re working with other people.
   2. There will be too many features in a complex system, and it will be impossible to check. In a group, the team can run their own Unit test to notify others for any changes. That way, one programmer’s code will not break another person’s work. It will also notify any broken program.

Bonus Points:

Create a repository on github and commit any file to it to demonstrate your ability to use Github. Include a link to the repository inside your test submission.