WEEK#2 - ASSIGNMENT 2

MS Connect

WIPRO
Applying Thought

Vijay Sekhar G

Objectives

- Get comfortable with User Interface Design and Client Side Scripting.
- Start thinking more carefully.
- Solve problems using HTML5, Java Script and JQuery.

Reasonable

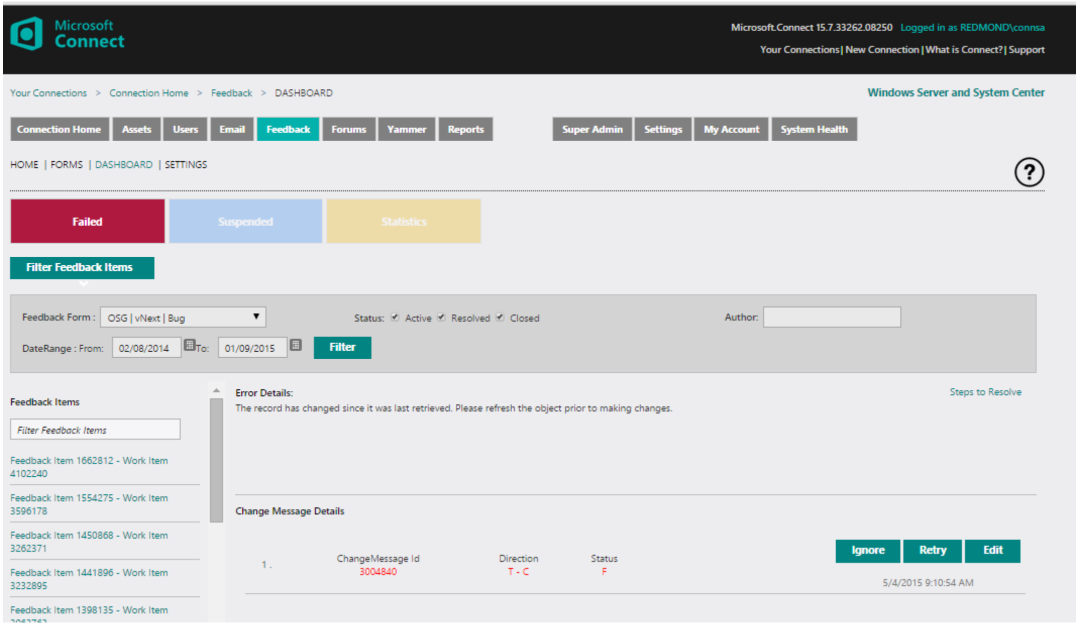
1. Communicating with colleagues about problem problems in English (or some other spoken language).
2. Discussing the assignment material with others in order to understand it better.
3. Helping a colleagues identify a bug in his or her code, as by viewing, compiling, or running his or her code, even on your own computer.
4. Incorporating snippets of code that you find online or elsewhere into your own code, provided that those snippets are not themselves solutions to assigned problems and that you cite the snippets' origins.
5. Sending or showing code that you've written to someone, possibly a colleagues, so that he or she might help you identify and fix a bug.

Deliverables

- Document your solutions in word file.
- Write proper comments for each line in your source code.
- Document the output of your program.
- Your program should address the problem, there should NOT be any deviations in output.

SECTION – A

1 Design the following static screen using HTML5 elements.


Microsoft
Connect
Microsoft.Connect 15.7.33262.08250 Logged in as REDMOND\connsa
Your Connections | New Connection | What is Connect? | Support
Your Connections > Connection Home > Feedback > DASHBOARD
Windows Server and System Center
Connection Home
Assets
Users
Email
Feedback
Forums
Yammer
Reports
Super Admin
Settings
My Account
System Health
HOME | FORMS | DASHBOARD | SETTINGS
?
Failed
Suspended
Statistics
Filter Feedback Items
Filter
Feedback Items
Filter Feedback Items
Steps to Resolve
Ignore
Retry
Edit

Feedback Form : OSG | vNext | Bug ▼

Status: ☑ Active ☑ Resolved ☑ Closed

Author:

DateRange : From: 02/08/2014 To: 01/09/2015

Feedback Item 1662812 - Work Item 4102240

Feedback Item 1554275 - Work Item 3596178

Feedback Item 1450868 - Work Item 3262371

Feedback Item 1441896 - Work Item 3232895

Feedback Item 1398135 - Work Item

Error Details:
The record has changed since it was last retrieved. Please refresh the object prior to making changes.

Change Message Details

	ChangeMessage Id	Direction	Status	
1 .	3004840	T - C	F	5/4/2015 9:10:54 AM

Your design must be better than this. Add images folder and add one or images to this. Display an image as logo in the header. Your styles must be in CSS. Do not use styles within HTML tags.

2 Design animated 3D object using HTML5 & CSS (E.g. Display Bouncing ball)

SECTION – B

1 Write a program in JavaScript

1. That prints a multiplication table for numbers up to 12.
2. That prints the next 20 leap years.
3. That reverses a list, preferably in place.
4. That checks whether an element occurs in a list.
5. That returns the elements on odd positions in a list.
6. That computes the running total of a list.
7. That combines two lists by alternatingly taking elements, e.g. [a,b,c], [1,2,3] -> [a,1,b,2,c,3].
8. That takes the duration of a year (in fractional days) for an imaginary planet as an input and produces a leap-year rule that minimizes the difference to the planet's solar year.
9. Given two strings, write a program that efficiently finds the longest common subsequence.
10. Store a set of words in an array and display the contents both forward and backward.
11. Create a *grades* object that stores a set of student grades in an object. Provide a function for adding a grade and a function for displaying the student's grade average.
12. Create an object that stores individual letters in an array and has a function for displaying the letters as a single word.

2 Write a function that inserts an element into a list only if the element to be inserted is larger than any of the elements currently in the list. Larger can mean either greater than when working with numeric values, or further down in the alphabet, when working with textual values.

3 Create a Person class that stores a person's name and their gender. Create a list of at least 10 Person objects. Write a function that displays all the people in the list of the same gender.

④ A stack can be used to ensure that an arithmetic expression has balanced parentheses. Write a function in JavaScript that takes an arithmetic expression as an argument and returns the position in the expression where a parenthesis is missing. An example of an arithmetic expression with unbalanced parentheses is
2.3 + 23 / 12 + (3.14159 * .24.

⑤ A postfix expression evaluator works on arithmetic expressions taking the following form:

op1 op2 operator

Using two stacks—one for the operands and one for the operators—design and implement a JavaScript function that converts infix expressions to postfix expressions, and then use the stacks to evaluate the expression.

⑥ Demonstrates how the priority queue system works using JavaScript.

⑦ Write a JavaScript program that uses a singly linked list to keep track of a set of test grades entered interactively into the program.

⑧ According to legend, the first-century Jewish historian Flavius Josephus was about to be captured along with a band of 40 compatriots by Roman soldiers during the Jewish-Roman War. The Jewish soldiers decided that they preferred suicide to being captured and devised a plan for their demise. They were to form a circle and kill every third soldier until they were all dead. Josephus and one other decided they wanted no part of this and quickly calculated where they needed to place themselves so they would be the last survivors. Write a program that allows you to place n people in a circle and specify that every *m* th person will be killed. The program should determine the number of the last two people left in the circle. Use a circularly linked list to solve the problem using Java Script.

⑨ Write a JavaScript program that takes a set of names and phone numbers from a text file (prepare a JSON file) and stores them in a Dictionary object. Include in your program the ability to display one phone number, display all phone numbers, add new phone numbers, remove phone numbers, and clear out the list of numbers.

⑩ Use linear probing to create a simple dictionary to store the definitions of words.
Your program should have two parts.
The first part reads a text file (prepare a JSON file) that contains a list of words and definitions and stores them in a hash table.
The second part of the program allows a user to enter a word and see the definition of that word.