



DESK No.

--	--	--

FAMILY NAME: \_\_\_\_\_

GIVEN NAMES: \_\_\_\_\_

SIGNATURE: \_\_\_\_\_

STUDENT NUMBER:

--	--	--	--	--	--	--

**SEMESTER 1, 2021 EXAMINATIONS****CITS3403  
Agile Web Development**

Physics, Mathematics & Computing  
EMS

This paper contains: 18 Pages (including title page)

Time Allowed: **2:00** hours**INSTRUCTIONS:**

This exam consists of FOUR questions worth 60 marks all together.  
Attempt all questions, and write your responses in the boxes provided.  
No notes, written materials, or electronic devices are allowed.  
Reference materials are provided for the exam.

**THIS IS A CLOSED BOOK EXAMINATION****SUPPLIED STATIONERY****ALLOWABLE ITEMS****PLEASE NOTE**

*Examination candidates may only bring authorised materials into the examination room. If a supervisor finds, during the examination, that you have unauthorised material, in whatever form, in the vicinity of your desk or on your person, whether in the examination room or the toilets or en route to/from the toilets, the matter will be reported to the head of school and disciplinary action will normally be taken against you. This action may result in your being deprived of any credit for this examination or even, in some cases, for the whole unit. This will apply regardless of whether the material has been used at the time it is found.*

*Therefore, any candidate who has brought any unauthorised material whatsoever into the examination room should declare it to the supervisor immediately. Candidates who are uncertain whether any material is authorised should ask the supervisor for clarification.*

*Candidates must comply with the Examination Rules of the University and with the directions of supervisors.*

*No electronic devices are permitted during the examination.*

*All question papers and answer booklets are the property of the University and remain so at all times.*

This page has been left intentionally blank

1. (15 marks)
  - a. (5 Marks) Explain how a modern web browser uses the technologies, HTML, CSS and JavaScript, to deliver a web application to a client.
  - b. (5 Marks) Give an example of a user story, and explain how user stories are used to manage requirements in an agile software development process.

- c. (5 Marks) Describe, with an example, how AJAX and the Document Object Model are used to request data from a web server and use it to dynamically update a web page.

2. (15 marks)

- a. (10 marks) Describe, as clearly as you can, the appearance and style of the page generated by the attached code when it is rendered by a modern web browser.

```
1 <html>
2 <head>
3   <meta charset='utf-8' />
4   <title>10 marks!</title>
5   <style>
6     body {color:white; background-color:black;}
7     p span.b {font-family:'monospace';}
8     li.d {font-weight:bold}
9     li.d ul { font-weight:normal;}
10    li.d > ul:hover {border:solid; margin:1em; padding:1em;}
11  </style>
12  <script>
13    function info(){
14      let dt = new Date();
15      document.getElementById('info').innerHTML=
16        'Hey ' + navigator.platform + ' user! Today is ' +
17        dt.getDate() + '/' + (dt.getMonth()+1) + '/' + dt.getFullYear() + '.'
18    }
19    window.onload = info;
20  </script>
21 </head>
22 <body>
23   <h1>Agile Web Development Exam, Question 2</h1>
24   <p> Flask is a micro-framework for building web applications.
25     The class <span class='b'>Flask</span> is actually a <span class='b'>webserver</span>,
26     that will listen for requests, and respond to those requests by executing python functions.
27   </p>
28   <div>
29     The structure of a typical <span class='b'>Flask</span> project is as follows:
30     <ul><li>app.py</li>
31       <li class='d'>app
32         <ul><li>models.py</li>
33           <li>routes.py</li>
34           <li class='d'>static
35             <ul><li>file.css</li>
36               <li>file.js</li>
37             </ul>
38           </li>
39           <li class='d'> templates
40             <ul><li>base.html</li></ul>
41           </li>
42         </ul>
43       </li>
44     </ul>
45   </div>
46   <div id='info'></div>
47 </body>
48 </html>
```

*Answer 2a here*

*Continue 2a here*

- b. (5 marks) Explain, with an example, how a closure works in JavaScript, and why closures are important when handling web requests.
-



3. (15 marks)

- a. (10 marks) Use flask-sqlalchemy to write a set of models satisfying the following specification for a movie review website:
- (a) There should be a model called **Movie**, which has a **title**, a **year\_of\_release**, and a **description**.
  - (b) There should be a model called **Reviewer**, which has a **username**, an **email**, and a short **biography**.
  - (c) There should be a model called **Review**, which has a **reviewer** (linking to the **Reviewer** who wrote the review), a **movie** (linking to the **Movie** the review is for), the **review\_text** describing the reviewer's opinion of the film, and a **rating** (the number of stars between one and five that the reviewer awarded the movie).
  - (d) The **Movie** model should also provide a function **star\_rating** that calculates the average number of stars awarded to the movie by reviewers, or 0 if the movie has no reviews yet.

As a prompt, the following models class is given as a template.

```
1  from app import db
2  from sqlalchemy.orm import relationship
3
4  class Student(db.Model):
5      __tablename__ = 'students'
6      id = db.Column(db.String(8), primary_key = True)
7      first_name = db.Column(db.String(64))
8      surname = db.Column(db.String(64))
9      project_id = db.Column(db.Integer, db.ForeignKey('projects.project_id'), nullable=True)
10
11     def get_partners(self):
12         if not self.project:
13             return []
14         team = self.project.team
15         team.remove(self)
16         return team
17
18     def __repr__(self):
19         return '[Number: {}, Name: {}]'.format(self.id, \
20             self.__str__())
21
22     class Project(db.Model):
23         __tablename__ = 'projects'
24         project_id = db.Column(db.Integer, primary_key = True)
25         description = db.Column(db.String(64))
26         team = relationship('Student', backref='project')
27
28         def __repr__(self):
29             return '[PID: {}, Desc: {}]'.format(\
30                 self.project_id, \
31                 self.description)
```

*Answer question 3a here*

*Continue question 3a here*

- b. (5 marks) Explain how *Test Doubles* are important in the context of:
- Test driven design
  - Repeatability of system tests
-

4. (15 marks)
  - a. (10 marks) Describe the functionality of your Web Application project, or a similar project that you are familiar with. Sketch a *Restful API* (include the routes and JSON schema) that would be able to provide the same functionality if this project was implemented as a *single page application*.

- b. (5 marks) Explain how the Representative State Architecture allows a web application to scale in response to both increased user demand and increased application functionality.



## HTML 5 NEW TAG

### TAG NOT SUPPORTED IN HTML 5

<b>&lt;!DOCTYPE&gt;</b>	Define a comment
<b>&lt;!DOCTYPE&gt;</b>	Defines the document type
<b>&lt;a&gt;</b>	Defines a hyperlink href, hreflang, media, ping, rel, target, type
<b>&lt;abbr&gt;</b>	Defines an abbreviation
<b>&lt;acronym&gt;</b>	Used to define an embedded acronym
<b>&lt;address&gt;</b>	Defines an address element
<b>&lt;applet&gt;</b>	Used to define an embedded applet
<b>&lt;area&gt;</b>	Defines an area inside an image map alt, coords, href, hreflang, media, ping, rel, shape, target, type
<b>&lt;article&gt;</b>	Defines an article cite, pubdate
<b>&lt;aside&gt;</b>	Defines content aside from the page content
<b>&lt;audio&gt;</b>	Defines sound content autobuffer, autoplay, controls, src
<b>&lt;b&gt;</b>	Defines bold text
<b>&lt;base&gt;</b>	Defines a base URL for all the links in a page href, target
<b>&lt;basefont&gt;</b>	Used to define a default font-color, font-size, or font-family for all the document
<b>&lt;bdo&gt;</b>	Defines the direction of text display dir
<b>&lt;big&gt;</b>	Used to make text bigger
<b>&lt;blockquote&gt;</b>	Defines a long quotation cite
<b>&lt;body&gt;</b>	Defines the body element
<b>&lt;br&gt;</b>	Inserts a single line break Defines a push button autofocus, disabled, form, formaction, formenctype, formmethod, formnovalidate, formtarget, name, type, value
<b>&lt;button&gt;</b>	
<b>&lt;canvas&gt;</b>	Defines graphics height, width
<b>&lt;caption&gt;</b>	Defines a table caption Used to center align text and content
<b>&lt;center&gt;</b>	
<b>&lt;cite&gt;</b>	Defines a citation Defines computer code text autobuffer, autoplay, controls, src
<b>&lt;code&gt;</b>	
<b>&lt;col&gt;</b>	Defines attributes for table columns Defines groups of table columns span
<b>&lt;colgroup&gt;</b>	
<b>&lt;command&gt;</b>	Defines a command button checked, disabled, icon, label, radiogroup, type

<b>&lt;datalist&gt;</b>	Defines a dropdown list
<b>&lt;dd&gt;</b>	Defines a definition description
<b>&lt;del&gt;</b>	Defines deleted text cite, datetime
<b>&lt;details&gt;</b>	Defines details of an element open
<b>&lt;dialog&gt;</b>	Defines a dialog (conversation)
<b>&lt;dfn&gt;</b>	Defines a definition term
<b>&lt;dir&gt;</b>	Used to define a directory list
<b>&lt;div&gt;</b>	Defines a section in a document
<b>&lt;dl&gt;</b>	Defines a definition list
<b>&lt;dt&gt;</b>	Defines a definition term
<b>&lt;em&gt;</b>	Defines emphasized text
<b>&lt;embed&gt;</b>	Defines external interactive content or plugin height, src, type, width
<b>&lt;fieldset&gt;</b>	Defines a fieldset disabled, form, name
<b>&lt;figure&gt;</b>	Defines a group of media content, and their caption
<b>&lt;font&gt;</b>	Used to define font face, font size, and font color of text
<b>&lt;footer&gt;</b>	Defines a footer for a section or page
<b>&lt;form&gt;</b>	Defines a form accept-charset, action, autocomplete, enctype, method, name, novalidate, target
<b>&lt;frame&gt;</b>	Used to define one particular window (frame) within a frameset
<b>&lt;frameset&gt;</b>	Used to define a frameset, which organized multiple windows (frames)
<b>&lt;h1&gt; to &lt;h6&gt;</b>	Defines header 1 to header 6
<b>&lt;head&gt;</b>	Defines information about the document
<b>&lt;header&gt;</b>	Defines a header for a section or page
<b>&lt;hgroup&gt;</b>	Defines information about a section in a document
<b>&lt;hr&gt;</b>	Defines a horizontal rule
<b>&lt;html&gt;</b>	Defines an html document manifest, xhtml
<b>&lt;i&gt;</b>	Defines italic text
<b>&lt;iframe&gt;</b>	Defines an inline sub window height, name, sandbox, seamless, src, width
<b>&lt;img&gt;</b>	Defines an image alt, src, height, ismap, usemap, width
<b>&lt;input&gt;</b>	Defines an input field accept, alt, autocomplete, autofocus, checked, disabled, form, formaction, formenctype, formmethod, formnovalidate, formtarget, height, list, max, maxlength, min, multiple, name, pattern, placeholder, readonly, required, size, src, step, type, value, width

<b>&lt;ins&gt;</b>	Defines inserted text cite, datetime
<b>&lt;keygen&gt;</b>	Defines a generated key in a form autofocus, challenge, disabled, form, keytype, name
<b>&lt;kbd&gt;</b>	Defines keyboard text
<b>&lt;label&gt;</b>	Defines an inline sub window for, form
<b>&lt;legend&gt;</b>	Defines a title in a fieldset
<b>&lt;li&gt;</b>	Defines a list item value
<b>&lt;link&gt;</b>	Defines a resource reference href, hreflang, media, rel, sizes, type
<b>&lt;map&gt;</b>	Defines an image map name
<b>&lt;mark&gt;</b>	Defines marked text
<b>&lt;menu&gt;</b>	Defines a menu list label, type
<b>&lt;meta&gt;</b>	Defines meta information charset, content, http-equiv, name
<b>&lt;meter&gt;</b>	Defines measurement within a predefined range high, low, max, min, optimum, value
<b>&lt;nav&gt;</b>	Defines navigation links Used to display text for browsers that do not handle frames
<b>&lt;noframes&gt;</b>	
<b>&lt;noscript&gt;</b>	Defines a noscript section
<b>&lt;object&gt;</b>	Defines an embedded object data, form, height, name, type, usemap, width
<b>&lt;ol&gt;</b>	Defines an ordered list reversed, start
<b>&lt;optgroup&gt;</b>	Defines an option group label, disabled
<b>&lt;option&gt;</b>	Defines an option in a drop-down list disabled, label, selected, value
<b>&lt;output&gt;</b>	Defines some types of output for, form, name
<b>&lt;p&gt;</b>	Defines a paragraph
<b>&lt;param&gt;</b>	Defines a parameter for an object name, value
<b>&lt;pre&gt;</b>	Defines preformatted text
<b>&lt;progress&gt;</b>	Defines progress of a task of any kind max, value
<b>&lt;q&gt;</b>	Defines a short quotation cite
<b>&lt;rp&gt;</b>	Used in ruby annotations to define what to show browsers that do not support the ruby element
<b>&lt;rt&gt;</b>	Defines explanation to ruby annotations
<b>&lt;ruby&gt;</b>	Defines ruby annotations
<b>&lt;s&gt;, &lt;strike&gt;</b>	Used to define strikethrough text.



<b>&lt;video&gt;</b>	Defines a video autobuffer, autoplay, controls, height, loop, src, width
----------------------	---

## HTML5 TAG CHEAT SHEET

Created by [Websitesetup.org](https://websitesetup.org)

# CSS Cheat Sheet



## Syntax

```
/* Comments */  
@media type {  
  selector {  
    property: value;  
  }  
}
```

*Note:  
media type is optional*

### Inline Style

```
<tag style="property: value;">
```

### Embedded Style

```
<head>  
  <style type="text/css">  
    selector { property: value; }  
  </style>  
</head>
```

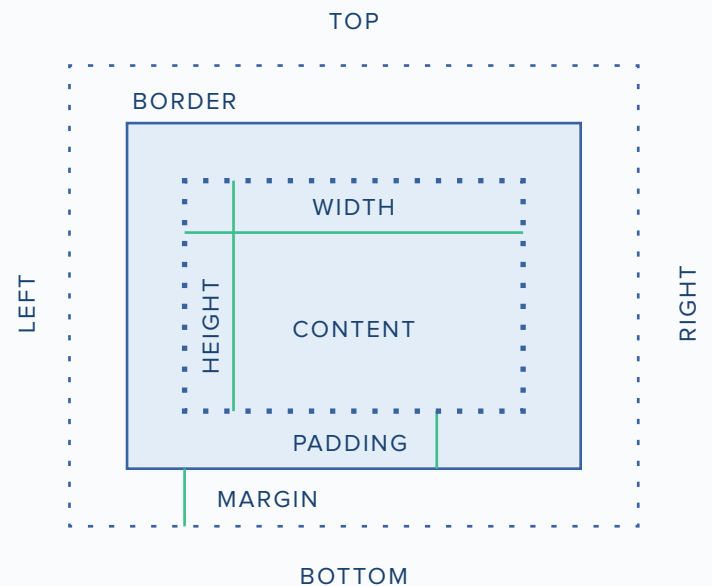
### External Style Sheet

```
<head>  
  <link rel="stylesheet"  
        type="text/css" href="style.css" />  
</head>
```

## Selectors

*	All elements
tag	All tag elements
tag *	All elements within tag
tag tag2	tag2 elements within tag
tag, tag2	All tag and tag2 elements
tag > tag2	tag2 is a child of tag
tag + tag2	tag2 preceded by tag
.class	Elements with class 'class'
tag.class	All tags with class 'class'
#id	Element with id 'id'
tag#id	Tag with id 'id'

## Box Model



## Boxes

margin	margin-top margin-right margin-bottom margin-left
padding	padding-top padding-right padding-bottom padding-left
border	border-top border-right border-bottom border-left
border-color	border-top-color border-right-color border-bottom-color border-left-color
border-style	border-top-style border-right-style border-bottom-style border-left-style
border-width	border-top-width border-right-width border-bottom-width border-left-width



Pseudo-selectors	
:active	Adds style to active element
:after	Adds content after element
:before	Ads content before element
:first-child	Adds style to first child
:first-letter	Adds style to first character
:first-line	Adds style to first line
:focus	Adds style to focused element
:hover	Adds style when mouse is over
:link	Adds style to unvisited link
:visited	Adds style to visited link
Text	
font-family	Specifies the font family
font-size	Specifies the font size
font-style	normal   italic   oblique   inherit
font-variant	normal   small-caps   inherit
font-weight	normal   bold   bolder   lighter   int (100- 900)   inherit
color	Sets the color of text
line-height	normal   int   %   inherit
text-align	left   right   center   justify   inherit
text-decoration	none   underline   overline   line-through   blink   inherit
text-indent	int   %   inherit
text-transform	none   capitalize   uppercase   lowercase   inherit
vertical-align	int   %   baseline   sub   super   top   text-top   middle   bottom   text-bottom   inherit
white-space	normal   nowrap   pre   pre-line   pre-wrap   inherit
word-spacing	normal   length   inherit

Size and colors	
Relative sizes	em   rem   ex   ch vw   vh   vmin   vmax   %
Absolute sizes	cm   mm   in   px   pt   pc
Colors	Hex #ff00ff RGB rgb(255,0,255)
Positioning	
position	absolute   fixed   relative   static   inherit
float	left   right   none   inherit
top, right, bottom, left	Sets the offset from the edge
clear	left   right   both   none   inherit ! fixed the document now
display	Specifies element placing in the document flow
overflow	visible   hidden   scroll   auto   inherit
visibility	visible   hidden   collapse   inherit
z-index	auto   int   inherit
Dimensions	
height	auto   int   %   inherit
max-height	none   int   %   inherit
max-height	none   int   %   inherit
max-width	none   int   %   inherit
min-height	int   %   inherit
min-width	int   %   inherit
width	auto   int   %   inherit
Other	
background	Format: background (color) (image) (repeat) (position)
cursor	Set the type of cursor
quotes	Set type of quotation marks

## Methods

### Object

toString  
toLocaleString  
valueOf  
hasOwnProperty  
isPrototypeOf  
propertyIsEnumerable

### String

charAt  
charCodeAt  
fromCharCode  
concat  
indexOf  
lastIndexOf  
localeCompare  
match  
replace  
search  
slice  
split  
substring  
substr  
toLowerCase  
toUpperCase  
toLocaleLowerCase  
toLocaleUpperCase

### RegExp

test  
match  
exec

### Array

concat  
join  
push  
pop  
reverse  
shift  
slice  
sort  
splice  
unshift

### Number

toFixed  
toExponential  
toPrecision

### Date

parse  
toString  
toISOString  
getDate  
getDay  
getFullYear  
getHours  
getMilliseconds  
getMinutes  
getMonth  
getSeconds  
getTime  
getTimezoneOffset  
getYear  
setDate  
setHours  
setMilliseconds  
setMinutes  
setMonth  
setSeconds  
setYear  
toLocaleTimeString

# JavaScript

## XMLHttpRequest

### Safari, Mozilla, Opera:

```
var req = new XMLHttpRequest();
```

### Internet Explorer:

```
var req = new  
ActiveXObject("Microsoft.XMLHTTP");
```

## XMLHttpRequest Object Methods

```
abort()  
getAllResponseHeaders()  
getResponseHeader(header)  
open(method, URL)  
send(body)  
setRequestHeader(header, value)
```

## XMLHttpRequest Object Properties

```
onreadystatechange  
readyState  
responseText  
responseXML  
status  
statusText
```

## XMLHttpRequest readyState Values

```
0    Uninitiated  
1    Loading  
2    Loaded  
3    Interactive  
4    Complete
```

## JAVASCRIPT IN HTML

### External JavaScript File

```
<script type="text/javascript"  
src="javascript.js"></script>
```

### Inline JavaScript

```
<script type="text/javascript">  
<!--  
    // JavaScript Here  
//-->  
</script>
```

## Functions

### Window

```
alert  
blur  
clearTimeout  
close  
focus  
open  
print  
setTimeout
```

### Built In

```
eval  
parseInt  
parseFloat  
isNaN  
isFinite  
decodeURI  
decodeURIComponent  
encodeURIComponent  
encodeURIComponent  
escape  
unescape
```

## REGULAR EXPRESSIONS - FORMAT

Regular expressions in JavaScript take the form:

```
var RegEx = /pattern/modifiers;
```

## REGULAR EXPRESSIONS - MODIFIERS

```
/g    Global matching  
/i    Case insensitive  
/s    Single line mode  
/m    Multi line mode
```

## REGULAR EXPRESSIONS - PATTERNS

```
^      Start of string  
$      End of string  
.      Any single character  
(a|b)  a or b  
(...)  Group section  
[abc]  Item in range (a or b or c)  
[^abc] Not in range (not a or b or c)  
a?     Zero or one of a  
a*     Zero or more of a  
a+     One or more of a  
a{3}   Exactly 3 of a  
a{3,}  3 or more of a  
a{3,6} Between 3 and 6 of a  
!(pattern) "Not" prefix. Apply rule when  
          URL does not match pattern.
```

## EVENT HANDLERS

```
onAbort      onMouseDown  
onBlur       onMouseMove  
onChange     onMouseOut  
onClick      onMouseOver  
onDbClick    onMouseUp  
onDragDrop   onMove  
onError      onReset  
onFocus      onResize  
onKeyDown    onSelect  
onKeyPress   onSubmit  
onKeyUp      onUnload  
onLoad
```

## FUNCTIONS AND METHODS

A method is a type of function, associated with an object. A normal function is not associated with an object.

Available free from  
AddedBytes.com

## DOM Methods

### Document

```
clear  
createDocument  
createDocumentFragment  
createElement  
createEvent  
createEventObject  
createRange  
createTextNode  
getElementsByName  
getElementById  
write
```

### Node

```
addEventListener  
appendChild  
attachEvent  
cloneNode  
createTextRange  
detachEvent  
dispatchEvent  
fireEvent  
getAttributeNS  
getAttributeNode  
hasChildNodes  
hasAttribute  
hasAttributes  
insertBefore  
removeChild  
removeEventListener  
replaceChild  
scrollIntoView
```

### Form

```
submit
```

### DOM Collections

```
item
```

### Range

```
collapse  
createContextualFragment  
moveEnd  
moveStart  
parentElement  
select  
setStartBefore
```

### Style

```
getPropertyValue  
setProperty
```

### Event

```
initEvent  
preventDefault  
stopPropagation
```

### XMLSerializer

```
serializeToString
```

### XMLHTTP

```
open  
send
```

### XMLDOM

```
loadXML
```

### DOMParser

```
parseFromString
```