CS1101S

STUDIO SESSION

SELF INTRO :)

Name

JC / Poly

CS Background

Fun Fact

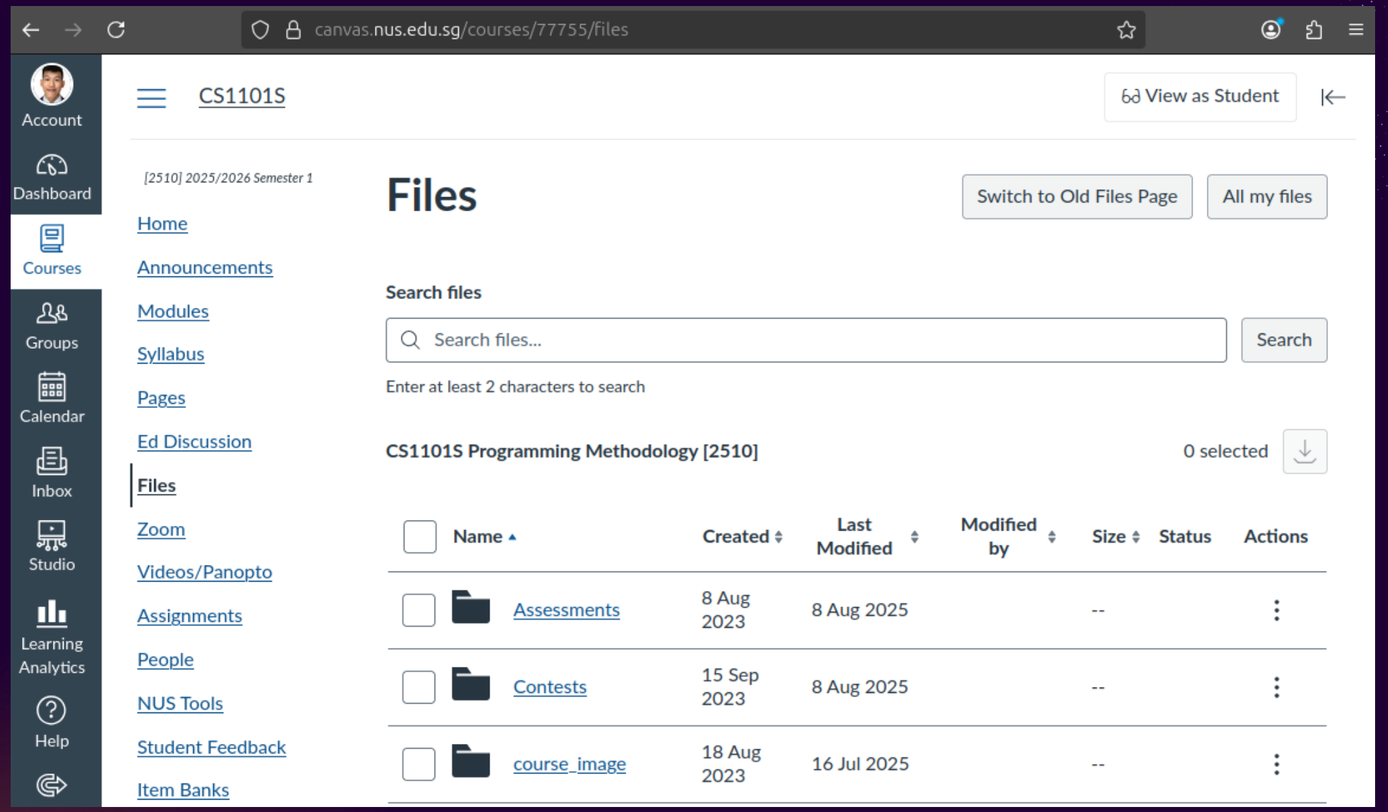
STUDIO TELEGRAM GROUP CHAT

For everyone in the studio group.

Please refrain from sharing, thanks :)

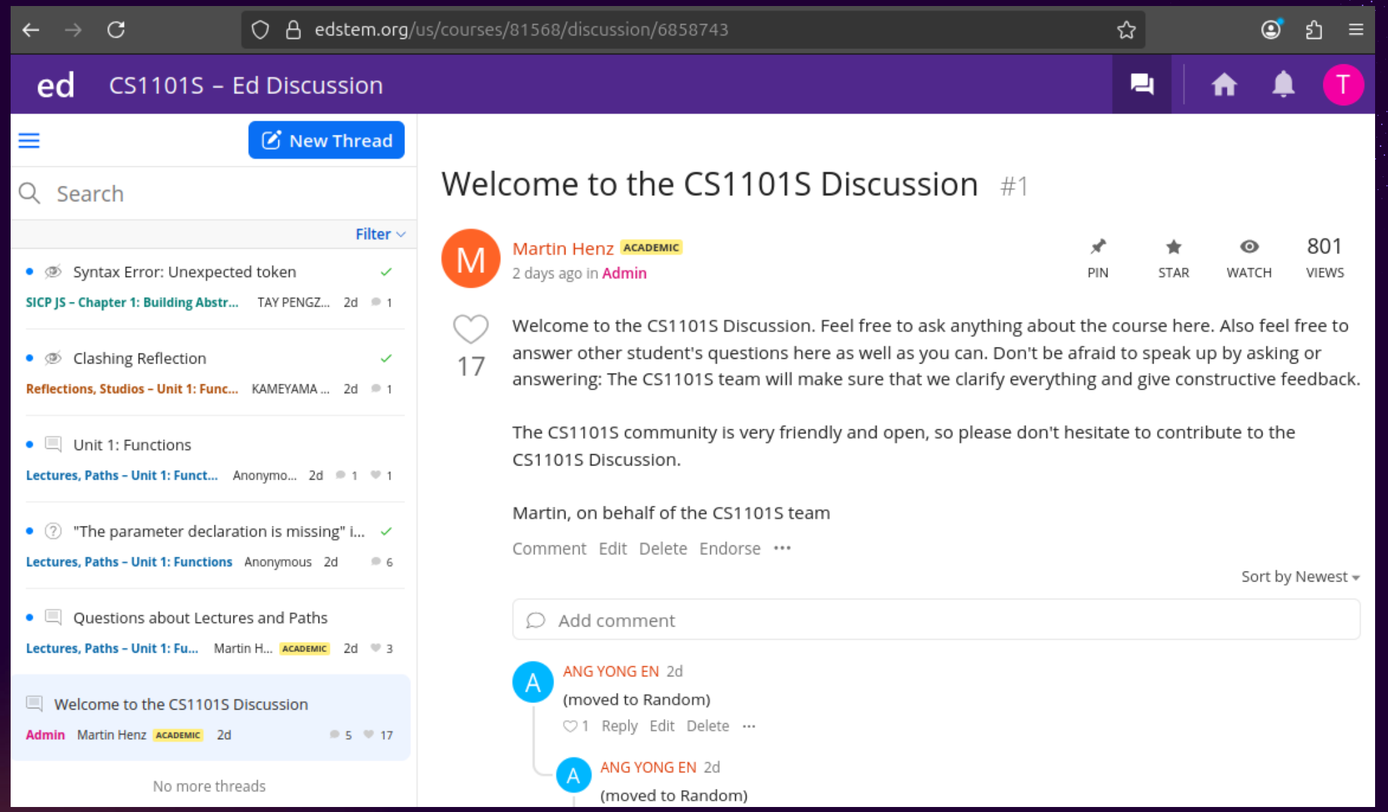
CANVAS

For class materials



ED FORUM

For class discussions



HONOR CODES

Essentially, Do The Practices For Your Own Learning

ACADEMIC DISHONESTY

The University and our course takes a strict view of cheating in any form.

- Any student who is found to have engaged in such misconduct will be subjected to disciplinary action by the University.
- Plagiarism. The practice of taking someone else's work or ideas and passing them off as one's own (e.g. copying from your classmates, seniors, books and/or online resources).

USE OF AI

When can and cannot use AI

REFLECTION

Preparation: Highly Encouraged
Try to explain your solution without looking at notes

STUDIO

Preparation: Highly Discouraged
During: Prohibited
Try to DIY an answer by yourself. Apply what your have learn, and identify what you do not know

MISSION & QUEST

During: Highly Discouraged
Learn to debug your codes. If you use AI, include a link in your code as a comment

MASTERY CHECKS

Preparation: Highly Encouraged
During: Prohibited
Ensure that you understand the topic well ahead of exams :)

STUDIO EXPECTATIONS

Minimumly try first ah...., if not be good enough to think on the spot for the answers. I’m here to be humbled. :)

But if you have any difficulty, note down what you struggled with, then in class figure out where the issue lie in. = productive studio :)

Have fun and we fast game settle :D

LECTURE SUMMARY

Primitive Expressions

Operator Combinations

Evaluating Combintations

Naming Abstraction

Functional Abstraction

Predicates and Conditional Expressions

PRIMITIVE EXPRESSIONS

Primitives (Numerals): 0, -42, 486

Combination: 486;

Means of Abstraction

OPERATOR COMBINATIONS

Operators: + - / *

Operands: Primitves

Combinations: 1 + 2

EVALUATING COMBINATIONS

Follows PERMDAS, Left to Right

(1 + 3 * 6) * (4 / 2)

= (1 + 18) * 2

= 19 * 2

= 38

MEANS OF ABSTRACTION

Naming

```
1  const x = 1;
2  x * 2;
```

MEANS OF ABSTRACTION

Compound Functions

```
1 function square(x) {
2     return x * x;
3 }
4 square(12);
```

PREDICATES AND CONDITIONAL EXPRESSIONS

In the form: *predicate* ? *consequent-expr* : *alternate-expr*

```
1 function absolute(x) {
2     return x >= 0 ? x : -x;
3 }
4 absolute(-3);
```

X % Y

Gets Remainder of X divide Y; Follows sign of X

X % Y
EXAMPLES

```
1  // Type your program in here!
2  display(17 % 5);
3  display(17 % -5);
4  display(-17 % 5);
5  display(-17 % -5);
```

X % Y
EXAMPLES

```
1  // Type your program in here!
2  display(17 % 5);
3  display(17 % -5);
4  display(-17 % 5);
5  display(-17 % -5);
```

```
2
2
-2
-2
-2
```

PRE-DECLARED FUNCTIONS

Math Pre-declared Functions:
math_floor(x): ⌊3.7⌋ = 3
math_ceil(x): ⌈3.14⌉ = 4
math_abs(x): |-1| = 1
math_sqrt(x): √9 = 3

PRE-DECLARED FUNCTIONS

```
1  display(math_floor(3.7));
2  display(math_ceil(3.14));
3  display(math_abs(-1));
4  display(math_sqrt(9));
```

```
3
4
1
3
3
```

math_floor(x)

Test Your Understanding

```
1  display(math_floor(-1.5));
```

math_floor(x)

Test Your Understanding Solution

```
1  display(math_floor(-1.5));
```

```
-2
-2
```

</> Eval Clear

STUDIO SHEET 2

Now’s your turn!