Hello Class!
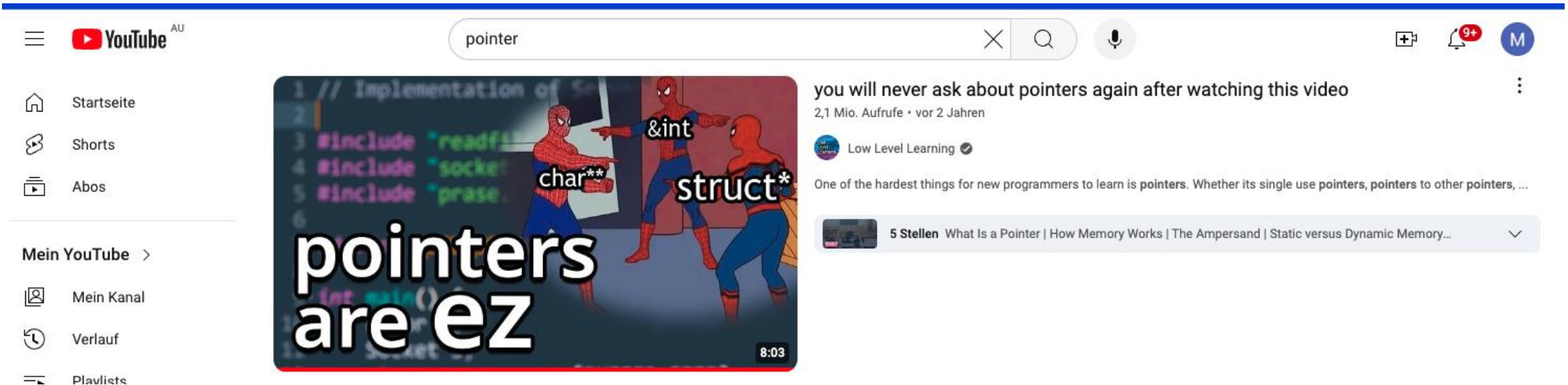
If you aren't familiar with **pointers** or would like a quick refresher, please watch the following video:

https://www.youtube.com/watch?v=2ybLD6_2gKM

global variable called *x*

```c
int x = 3;

int main(int argc, char **argv) {
    printf("%d\n", x); // Output 1
    x = 7;
    int x = 4;
    printf("%d\n", x); // Output 2
    x = 8;
    test();
    return 0;
}

void test(void) {
    printf("%d\n", x);
    return;
}
```

update global variable *x*

new local variable called *x*
! this variable now **shadows** the global *x* !

update local variable *x*

access global *x*

Pointers

int cents	define a variable called *cents*

scanf("%d", &cents)
 	
       → the address of the variable
           cents
	store the input at the address of *cents*

int *cents	define a pointer called *cents* (not initialised)
int try_one_coin(int *cents) function	defining a function called *try_one_coin*: create a pointer called cents it will point at whatever is passed to the function as input
*cents	using inside a function: access the value stored at the address cents is pointing to

Pointers

```c
#include <stdio.h>

void update_value(int *num) {
    *num = 42;  // Change the value at the memory address num is pointing to
}

int main() {
    int x = 10;
    printf("Before: %d\n", x);

    update_value(&x);  // Pass the address of x

    printf("After: %d\n", x);  // x is now changed
    return 0;
}
```

int cents	define a variable called *cents*

scanf("%d", &cents)
      → the address of the variable
        cents
	store the input at the address of *cents*

int *cents	<u>define a pointer</u> called *cents* (not initialised)
int try_one_coin(int *cents) function	<u>defining a function</u> called *try_one_coin*: create a pointer called cents it will point at whatever is passed to the function as input
*cents	<u>using inside a function</u>: access the value stored at the address cents is pointing to