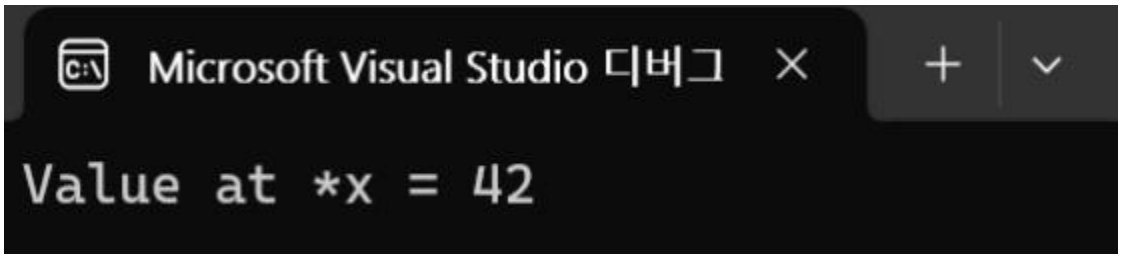
2019313550_박병현

실습 1.

```c
#include <stdio.h>
#include <stdlib.h>

int main() {
	int* x; // allocate the pointers x

	x = malloc(sizeof(int)); //Aloocate an int pointee, and set x to point ot it

	*x = 42; //Derefeence x to store 42 in its pointee

	printf("Value at *x = %d \n", *x);

	free(x);
	return 0;
}
```

실습 1 실행 화면.

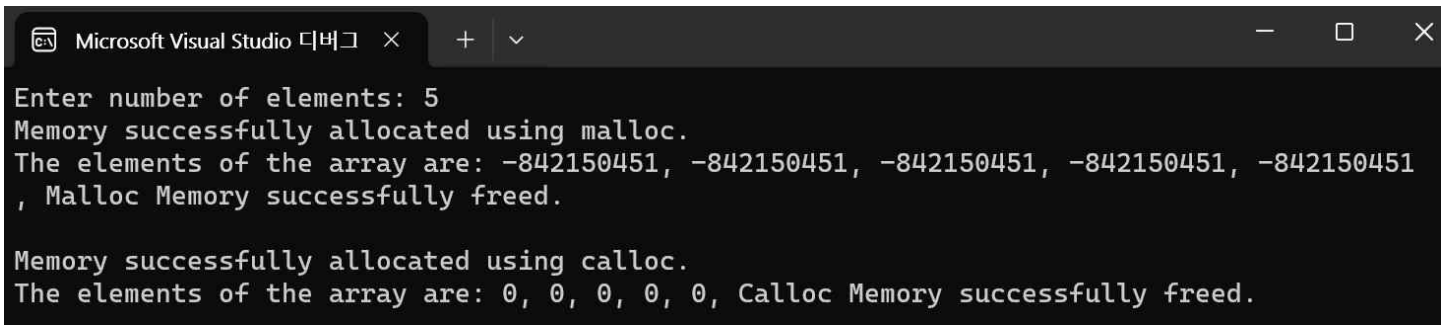


2019313550_박병현

실습 2.

```c
#include <stdio.h>
#include <stdlib.h>
int main() {
	// This pointer will hold the
	// base address of the block created
	int* ptr, * ptr1;
	int n, i;
	// Get the number of elements for the array
	n = 5;
	printf("Enter number of elements: %d\n", n);
	// Dynamically allocate memory using malloc()
	ptr = (int*)malloc(n * sizeof(int));
	// Dynamically allocate memory using calloc()
	ptr1 = (int*)calloc(n, sizeof(int));
	// Check if the memory has been successfully
	// allocated by malloc or not
	if (ptr == NULL || ptr1 == NULL) {
		printf("Memory not allocated.\n");
		exit(0);
	}
	else {
		// Memory has been successfully allocated
		printf("Memory successfully allocated using malloc.\n");
		// Print the elements of the array
		printf("The elements of the array are: ");
		for (i = 0; i < n; ++i) printf("%d, ", ptr[i]);
		free(ptr); // Free the memory
		printf("Malloc Memory successfully freed.\n");
		// Memory has been successfully allocated
		printf("\nMemory successfully allocated using calloc.\n");
		// Print the elements of the array
		printf("The elements of the array are: ");
		for (i = 0; i < n; ++i) printf("%d, ", ptr1[i]);
		free(ptr1); // Free the memory
		printf("Calloc Memory successfully freed.\n");
	}
	return 0;
}
```

실습 2 실행 화면.

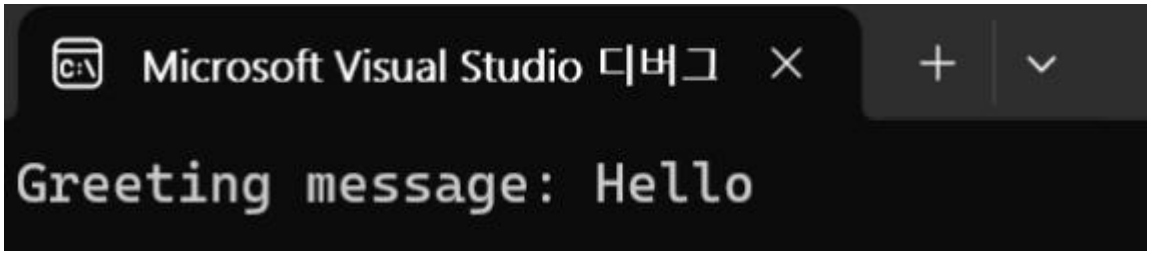


2019313550_박병현

실습 3.

```
#include <stdio.h>

int main() {
	char greeting[] = { 'H', 'e', 'l','l','o','\0' };
	//char greeting[]= "Hello";
	printf("Greeting message: %s\n", greeting);
	return 0;
}
```

실습 3 실행 화면.

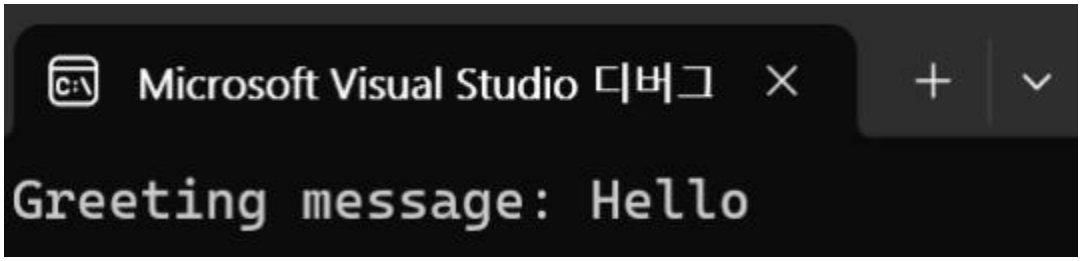


2019313550_박병현

실습 4.

```
#include <stdio.h>

int main() {
	//char greeting[]= "Hello";
	char* greeting = "Hello";
	printf("Greeting message: %s\n", greeting);
	return 0;
}
```

실습 4 실행 화면.

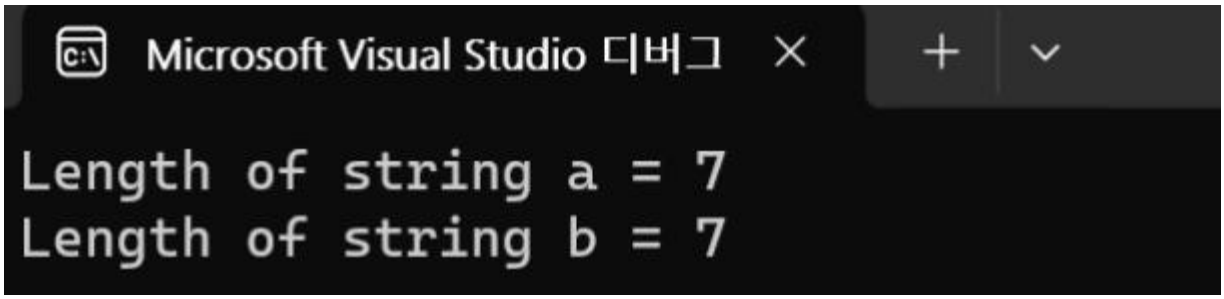


2019313550_박병현

실습 5.

```
#include<stdio.h>

int main() {
    char a[20] = "Program";
    char b[20] = { 'P','r','o','g','r','a','m','\0' };
    //using the %zu format specifier to print size_t;
    printf("Length of string a = %zu \n", strlen(a));
    printf("Length of string b = %zu \n", strlen(b));
    return 0;
}
```

실습 5 실행 화면.

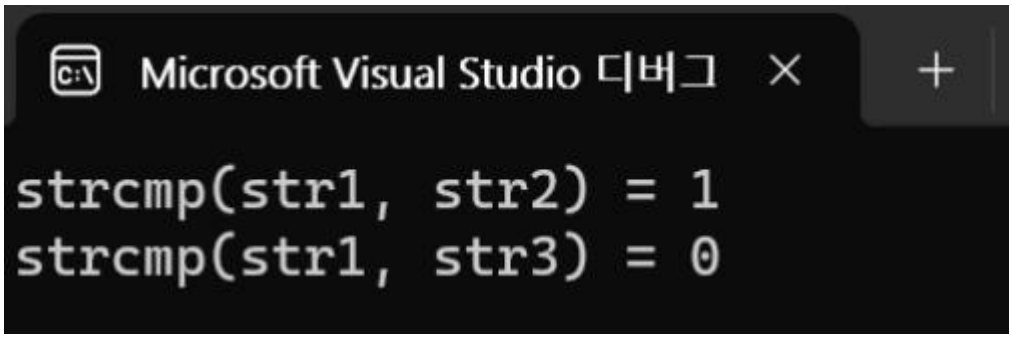


2019313550_박병현

실습 6.

```c
#include <stdio.h>
#include <string.h>

int main() {
    char str1[] = "abcd", str2[] = "abCd", str3[] = "abcd";
    int result;

    // comparing strings str1 and str2
    result = strcmp(str1, str2);
    printf("strcmp(str1, str2) = %d\n", result);
    // comparing strings str1 and str3
    result = strcmp(str1, str3);
    printf("strcmp(str1, str3) = %d\n", result);

    return 0;

}
```

실습 6 실행 화면.

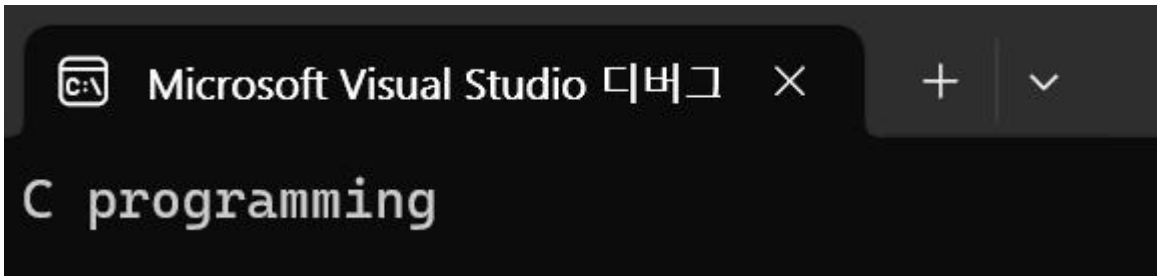


2019313550_박병현

실습 7.

```c
#include<stdio.h>
#include<string.h>

int main() {
	char str1[20] = "C programming";
	char str2[20];

	// comping str1 to str2
	strcpy_s(str2, 20, str1);
	//strcpy(sstr2, str1);
	puts(str2);// C programming

	return 0;
}
```

실습 7 실행 화면.

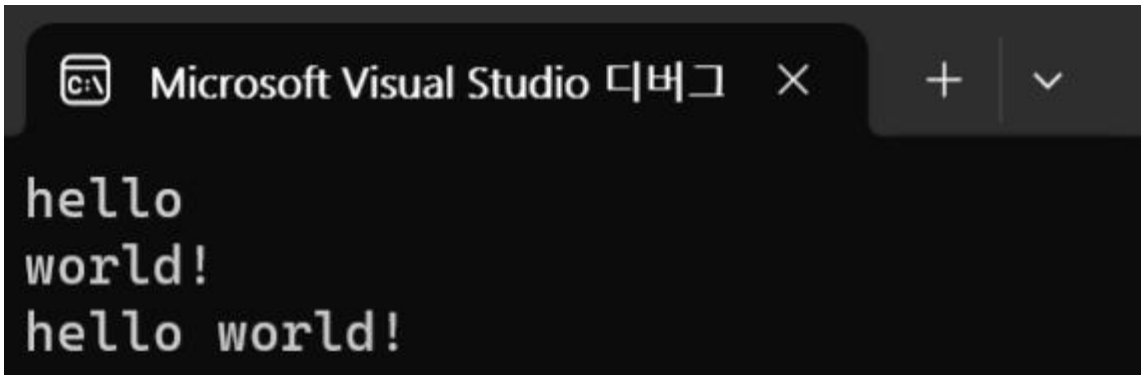


2019313550_박병현

실습 8.

```
#include<stdio.h>
#include<string.h>

int main() {
	char str1[14];
	strcpy_s(str1, 14, "hello ");
	printf("%s\n", str1);

	char str2[7];
	strcpy_s(str2, 7, "world!");
	printf("%s\n", str2);

	strcat_s(str1, 14, str2);
	printf("%s\n", str1);

	return 0;
}
```

실습 8 실행 화면.

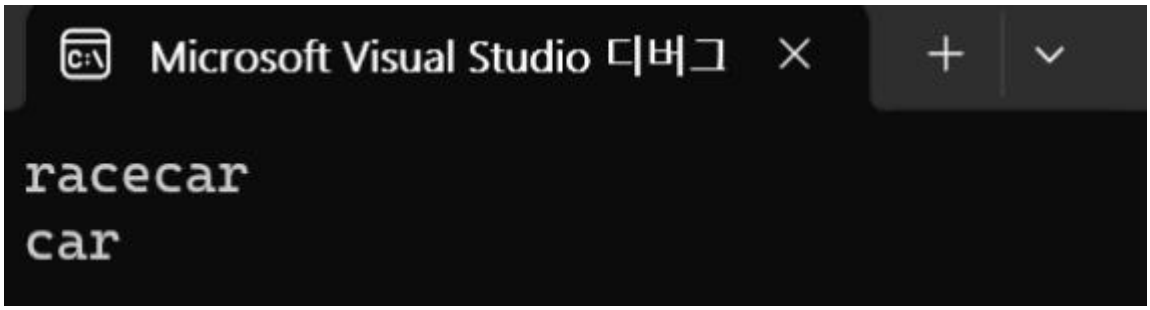


2019313550_박병현

실습 9.

```
#include<stdio.h>
#include<string.h>

int main() {
	char chars[8];
	strcpy_s(chars, 8, "racecar");
	char* str1 = chars;
	char* str2 = chars + 4;
	printf("%s\n", str1);
	printf("%s\n", str2);
	return 0;
}
```

실습 9 실행 화면.



2019313550_박병현

실습 10.

```
#include<stdio.h>

int main() {
    char line[150];
    printf("Enter a string: ");
    fgets(line, sizeof(line), stdin); //take input

    for (int i = 0, j; line[i] != '\0'; ++i) {
        //enter the loop if the character is not an alphabet and not the null character
        while (!(line[i] >= 'a' && line[i] <= 'z') && !(line[i] >= 'A' && line[i] <= 'Z') && !(line[i] == '\0')) {
            for (j = i; line[j] != '\0'; ++j) {
                // if jth element of lin ei s not an alphabet,
                // assign the value of (j+1)th element to the jth element
                line[j] = line[j + 1];
            }
            line[j] = '\n';
        }
    }
    printf("Output String: ");
    puts(line);
    return 0;
}
```

실습 10 실행 화면.