

PART 13-1. 변수 사용 영역

◆ 연습문제 ◆

1. 다음 문장에서 설명이 틀린 것을 모두 고르세요.

- ① 지역 변수는 선언된 함수가 반환되면 할당된 저장 공간이 자동으로 회수된다.
- ② 전역 변수와 정적 변수는 프로그램이 실행되는 동안 계속 저장 공간을 유지한다.
- ③ 정적 지역 변수와 전역 변수는 변수의 사용 범위가 같다.
- ④ auto를 사용한 지역 변수는 저장 공간 할당과 동시에 0으로 자동 초기화된다.
- ⑤ 하나의 함수 내에서는 같은 이름의 변수를 선언할 수 없다.
- ⑥ 전역 변수와 지역 변수는 같은 이름을 사용할 수 있으나 전역 변수에 우선권이 있다.

정답 : ③ 정적 지역 변수와 전역 변수는 변수의 사용 범위가 같다.

④ auto를 사용한 지역 변수는 저장 공간 할당과 동시에 0으로 자동 초기화된다.

⑤ 하나의 함수 내에서는 같은 이름의 변수를 선언할 수 없다.

⑥ 전역 변수와 지역 변수는 같은 이름을 사용할 수 있으나 전역 변수에 우선권이 있다.

해설 : 없음

2. 다음 프로그램의 실행 결과를 적어 보세요.

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

```
void func(void);
```

```
int a = 10;
```

```
int main(void) {  
    a = 20;  
    func();  
    printf("%d", a);  
  
    return 0;  
}
```

```
void func(void) {  
    a = 30;  
}
```

실행 결과

30

해설 : 없음

3. 실행 결과와 일치하도록 빈 칸에 적당한 예약어를 채우세요.

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

```
int main(void) {  
    int i, sum = 0;  
  
    for (i = 0; i < 10; i++) {  
        sum += func();  
    }  
    printf("%d", sum);  
  
    return 0;  
}
```

```
int func(void) {
```

```
(      ) int a = 0;

a++;

return a;
}
```

실행 결과

55

정답 : static

해설 : 없음

4. 다음 프로그램의 실행 결과를 예상해 보세요.

```
#include <stdio.h>

int a;

int main(void) {
    int b = 10; {
        static int a = 20;

        b = a;
    }
    a += b;

    printf("%d", a);

    return 0;
}
```

정답 : 20

해설 : 없음

5. 다음은 배열의 값을 차례로 반환하는 함수를 호출하여 배열의 모든 값을 더하는 프로그램입니다. `get_next` 함수를 작성하여 프로그램을 완성하세요.

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

```
double get_next(void);
```

```
double ary[] = {  
    0.1, 0.3, 0.5, 0.7, 0.9  
};
```

```
int main(void) {  
    double res = 0;
```

```
    int i;
```

```
    for (i = 0; i < 5; i++) {  
        res += get_next();  
    }
```

```
    printf("%.1lf, res");
```

```
    return 0;  
}
```

```
double get_next(void) {  
    (  
    )  
}
```

실행 결과

2.5

정답

```
double get_next(void) {  
    static int index = 0;  
    double val;  
  
    val = ary[index];  
    index++;  
  
    return val;  
}
```

해설 : 없음