

RCBScript Test Cases

Test Case 1: Correct Variable Declarations

File: test_var_decl.rcb

Program:

```
a:number = 42  
b:number = 3.14  
c:text = "Hello RCBScript"  
d:bool = yes  
e:char = 'Z'  
h = make handle<number>  
h = 99  
result("a = " + a)  
result("c = " + c)
```

Test Case 2: Type Error

File: test_type_error.rcb

Program:

```
x:number = 10  
x = "wrong" # ERROR: expected number, got text
```

Expected Compiler Output:

```
Line 2: Type Error → expected number, got text
```

Test Case 3: Handle Error

File: test_handle_error.rcb

Program:

```
p = make handle<number>  
discard q # ERROR: undeclared handle
```

Expected Compiler Output:

```
Line 2: Reference Error → discard called on undeclared handle
```

Test Case 4: Initialization Error

File: test_init_error.rcb

Program:

```
z:number  
result(z) # ERROR: variable used before assignment
```

Expected Compiler Output:

```
Line 2: Initialization Error → variable used before assignment
```

Test Case 5: Testing Framework Example

File: test_framework.rcb

Program:

```
func sum(a:number, b:number) -> number  
reply a + b
```

```
test "sum of integers"  
expect sum(2,3) is 5
```

```
test "variable type test"  
x:number = 5  
y:text = "hello"  
expect x is 5  
expect y is "hello"
```

Test Case 6: Function Declaration and Return Type Error

File: test_func_type_error.rcb

Program:

```
func greet(name:text) -> number  
reply "Hello " + name # ERROR: expected number return, got text
```

Expected Compiler Output:

```
Line 2: Type Error → function declared to return number, got text
```

Test Case 7: Scope Error

File: test_scope_error.rcb

Program:

```
func demo()
  x:number = 10

  result(x) # ERROR: x is out of scope
```

Expected Compiler Output:

```
Line 4: Scope Error → variable 'x' not visible outside function
```

Test Case 8: Duplicate Declaration Error

File: test_duplicate_decl.rcb

Program:

```
a:number = 100
a:text = "duplicate" # ERROR: variable redeclared with different
type
```

Expected Compiler Output:

```
Line 2: Declaration Error → variable 'a' redeclared with type text
(was number)
```

Test Case 9: Boolean Logic Error

File: test_bool_logic.rcb

Program:

```
flag:bool = yes
if flag + 1 # ERROR: invalid operation on bool
  result("ok")
```

Expected Compiler Output:

```
Line 2: Type Error → '+' not supported for bool and number
```

Test Case 10: Handle Reuse Error

File: test_handle_reuse.rcb

Program:

```
h = make handle<number>  
discard h  
result(h) # ERROR: handle already discarded
```

Expected Compiler Output:

```
Line 3: Handle Error → handle 'h' used after being discarded
```
