# Question:

**(a) Choose a programming language with which you are familiar and identify one of the semantic rules of that language. Give an example of a program in that language that violates that rule. Post a screen shot showing the error message that the compiler for that language would generate for your program example.**

Programming language chosen : **DART**

Dart is a statically-typed programming language, which means that all variables must be declared with a specific type before they are used. Additionally, in Dart, variables can be either nullable or non-nullable.

A nullable variable is a variable that can have a value of null or a valid value of its declared type. A non-nullable variable is a variable that must always have a valid value of its declared type and cannot be null.

One of the semantic rules of the Dart programming language is that all non-nullable variables must be initialized before they are used. This means that if you declare a non-nullable variable and do not assign a value to it, the Dart compiler will raise an error.

Here's an example program that violates this semantic rule:

# Dart Program:

```
void main() {

  int variable;          //integer variable is not initialized
  print(variable);

}
```

In this program, the integer variable `variable` is declared but not initialized. Since `variable` is declared as a non-nullable integer variable, the Dart compiler requires that `variable` be initialized with a valid integer value before it can be used.

*This program is syntactically correct but semantically wrong.*

When you try to run this program, the Dart compiler raises an error with a message that clearly indicates that `x` must be initialized before it is used. In the error message, the caret symbol (`^`) is pointing to the line where `x` is declared, indicating that the error occurred at that line.

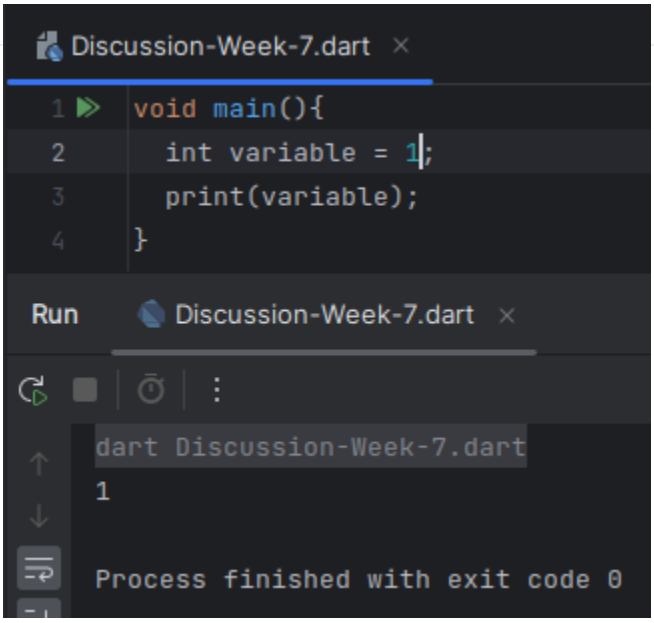
# Error:

```
dart Discussion-Week-7.dart
Discussion-Week-7.dart:3:9: Error: Non-nullable variable 'variable' must be assigned before it can be used.
  print(variable);
        ^^^^^^^^

Process finished with exit code 254
```

## Solution:

To fix this error, you can initialize `variable` with a valid integer value before using it, as shown in the corrected example program I provided earlier. This way, `variable` will have a valid value before it is used, and the program will run without any errors.



## (b) Then review the requirements for project 4. Select one of the semantic errors that you are to detect in project 4 and provide a test case that will detect it.

Semantic Error Chosen from Project 4:

*If Condition Not Boolean*

## Test Case:

```
-- Comprehensive test with nested if

function main a: integer, b: boolean, c: real returns integer;
   d: integer is 8;
   e: real is 3.75;
   f: boolean is true and not b;
begin
   if a > 5 and a < 1 and c = 5. or c /= 8.E4 or f or d + e then
      if c >= 7.E-2 and c <= 5.2 or false then
         a + 2 - 7.E+2 / 9 * 4;
      else
         a rem 2 - 5 / c;
      endif;
    else
       a ** 2 rem 3;
    endif;
end;
```

# Error:

The given test case fails the IF-Boolean rule because it contains a non-Boolean expression in the condition of the first *if* statement. Specifically, the expression $d + e$ is not a Boolean expression, as it evaluates to the value 11.75, which is a real number.

The IF-Boolean rule states that the condition of an *if* statement must be a Boolean expression, which means that it must evaluate to either *true* or *false*. In the given test case, the condition of the first if statement includes the non-Boolean expression $d + e$, which violates this rule.

If I convert this test case into Dart, we detect an error of IF condition not being a boolean error.

## Dart code:

Discussion-Week-7.dart

```dart
import 'dart:math';
void main(){
  function(2, true, 4);
}
void function(int a, bool b, double c) {
  int d = 8;
  double e = 3.75;
  bool f = true && !b;
  if (a > 5 && a < 1 && c == 5.0 || c != 8.0e4 || f || d + e) {
    if (c >= 7.0e-2 && c <= 5.2 || false) {
      a + 2 - 7.0e2 / 9 * 4;
    } else {
      a % 2 - 5 / c;
    }
  } else {
    if (a > 3) {
      if (b) {
        d + 2 * e;
      } else {
        a * 5;
      }
    } else {
      pow(a, 2) % 3;
    }
  }
}
```

## Error:

```
dart Discussion-Week-7.dart
Discussion-Week-7.dart:9:58: Error: A value of type 'double' can't be assigned to a variable of type 'bool'.
  if (a > 5 && a < 1 && c == 5.0 || c != 8.0e4 || f || d + e) {
                                                       ^
```

# References:


1. "Dart Language Tour." Dart, https://dart.dev/guides/language/language-tour#variables. Accessed 24 April 2023.
2. "Dart Programming Language Specification." Dart, https://dart.dev/guides/language/spec. Accessed 24 April 2023.
3. "Dart Static Types." Dart, https://dart.dev/guides/language/type-system. Accessed 24 April 2023.