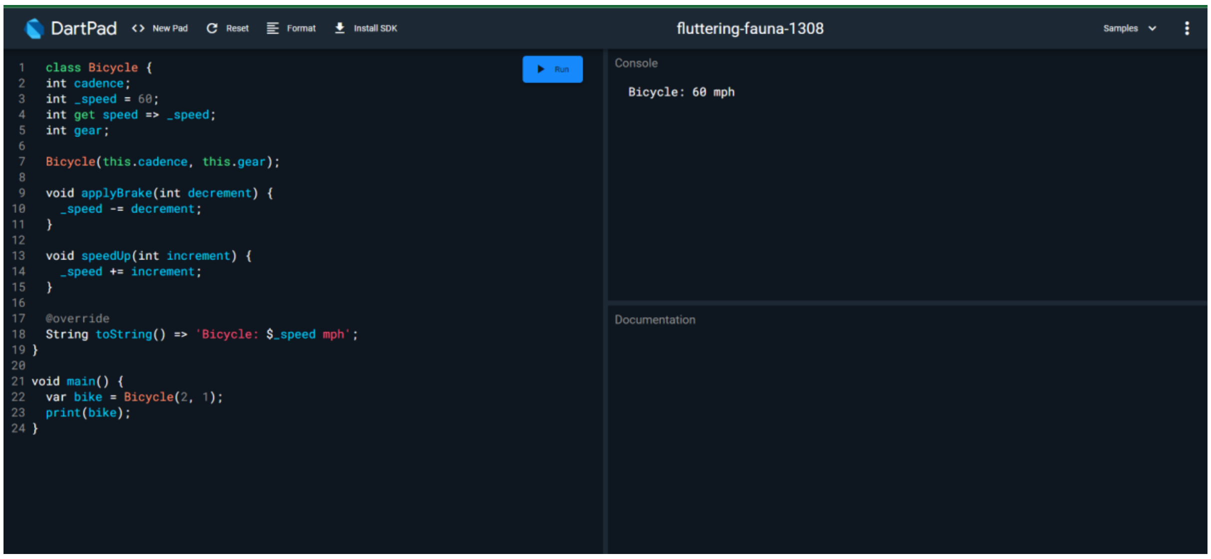
NAMA : RIJAL AHMAD JANANI

NPM : 065119062

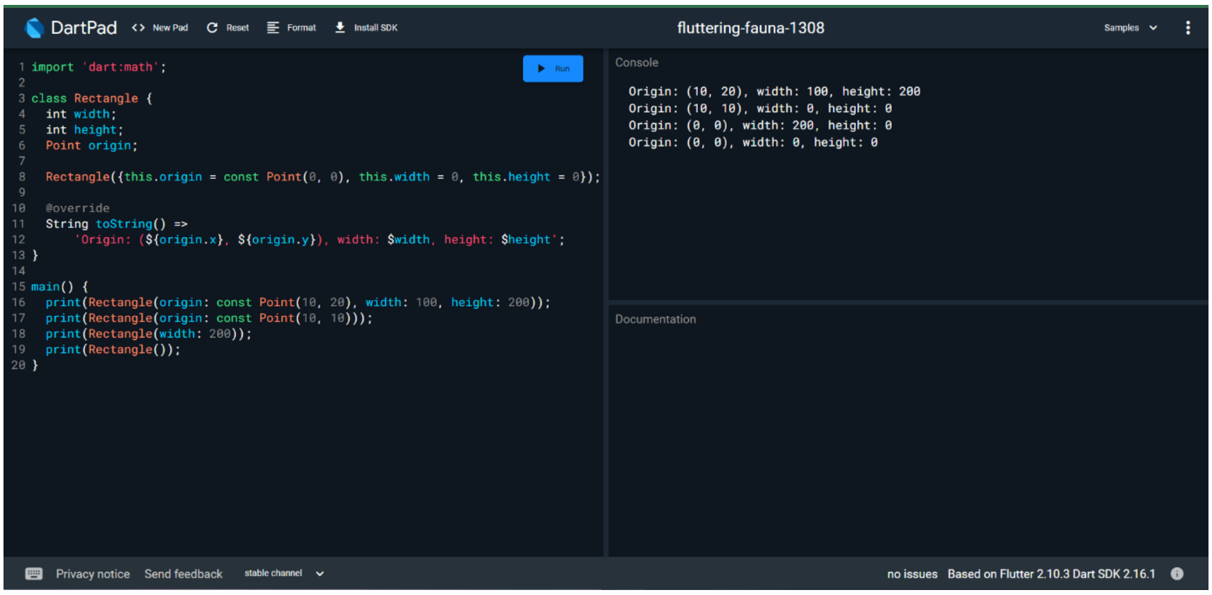
1. MEMBUAT CLASS DART SEDERHANA



```dart
class Bicycle {
  int cadence;
  int _speed = 60;
  int get speed => _speed;
  int gear;

  Bicycle(this.cadence, this.gear);

  void applyBrake(int decrement) {
    _speed -= decrement;
  }

  void speedUp(int increment) {
    _speed += increment;
  }

  @override
  String toString() => 'Bicycle: $_speed mph';
}

void main() {
  var bike = Bicycle(2, 1);
  print(bike);
}
```

```
Bicycle: 60 mph
```

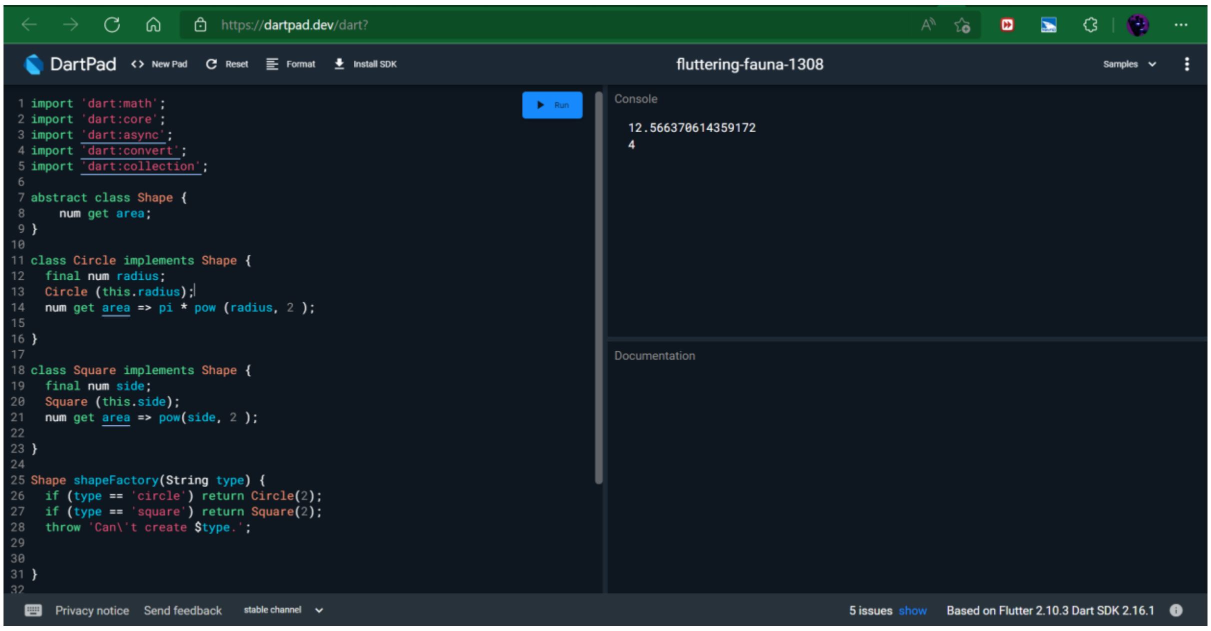
2. PARAMETER OPERASIONAL



```dart
import 'dart:math';

class Rectangle {
  int width;
  int height;
  Point origin;

  Rectangle({this.origin = const Point(0, 0), this.width = 0, this.height = 0});

  @override
  String toString() =>
      'Origin: (${origin.x}, ${origin.y}), width: $width, height: $height';
}

main() {
  print(Rectangle(origin: const Point(10, 20), width: 100, height: 200));
  print(Rectangle(origin: const Point(10, 10)));
  print(Rectangle(width: 200));
  print(Rectangle());
}
```

```
Origin: (10, 20), width: 100, height: 200
Origin: (10, 10), width: 0, height: 0
Origin: (0, 0), width: 200, height: 0
Origin: (0, 0), width: 0, height: 0
```

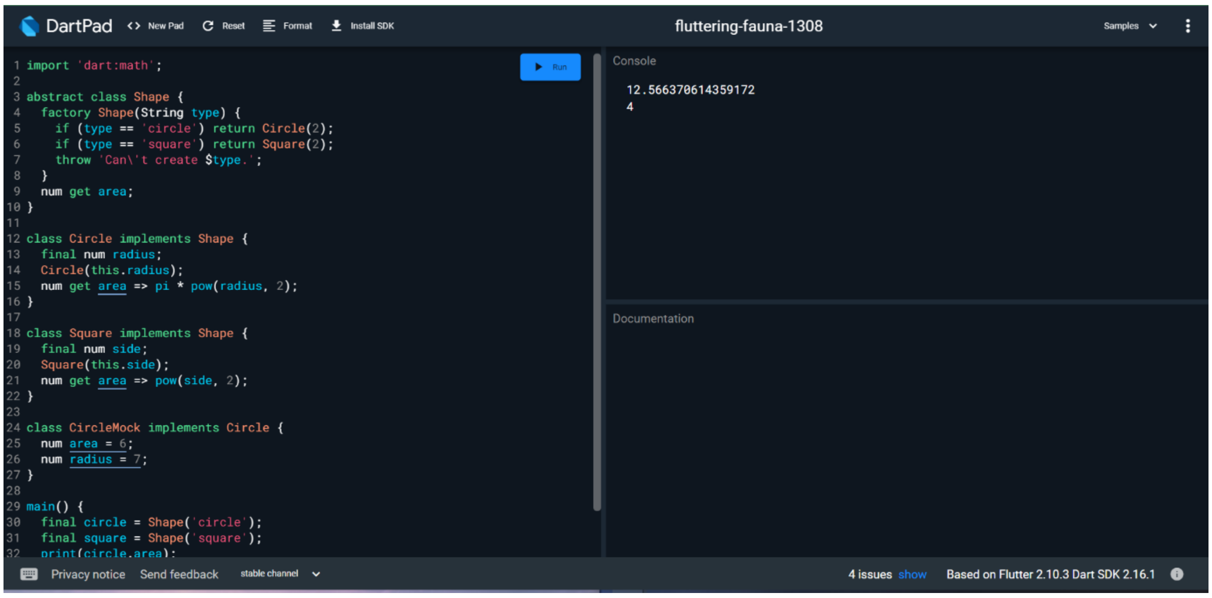
3. MEMBUAT PABRIK



```dart
import 'dart:math';
import 'dart:core';
import 'dart:async';
import 'dart:convert';
import 'dart:collection';

abstract class Shape {
    num get area;
}

class Circle implements Shape {
  final num radius;
  Circle (this.radius);
  num get area => pi * pow (radius, 2 );

}

class Square implements Shape {
  final num side;
  Square (this.side);
  num get area => pow(side, 2 );

}

Shape shapeFactory(String type) {
  if (type == 'circle') return Circle(2);
  if (type == 'square') return Square(2);
  throw 'Can\'t create $type.';


}

```

```
12.566370614359172
4
```

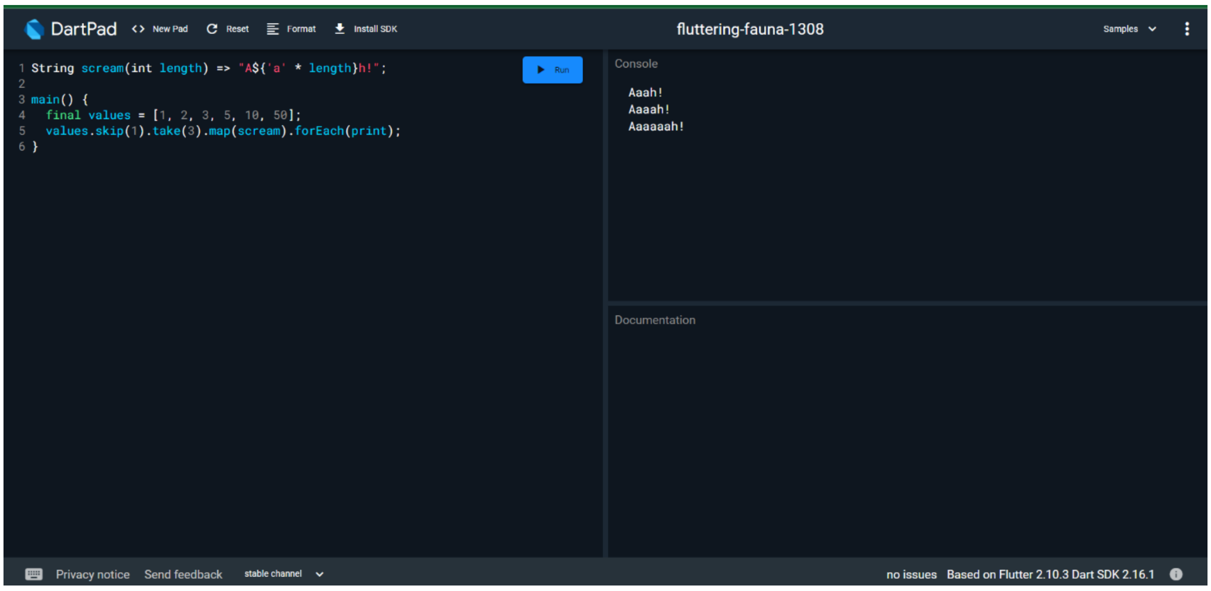
4. MENETAPKAN ANTAR MUKA



```dart
import 'dart:math';

abstract class Shape {
  factory Shape(String type) {
    if (type == 'circle') return Circle(2);
    if (type == 'square') return Square(2);
    throw 'Can\'t create $type.';
  }
  num get area;
}

class Circle implements Shape {
  final num radius;
  Circle(this.radius);
  num get area => pi * pow(radius, 2);
}

class Square implements Shape {
  final num side;
  Square(this.side);
  num get area => pow(side, 2);
}

class CircleMock implements Circle {
  num area = 6;
  num radius = 7;
}

main() {
  final circle = Shape('circle');
  final square = Shape('square');
  print(circle.area);
```

```
12.566370614359172
4
```

5. PEMEGRORAMAN FUNGSIONAL



```dart
String scream(int length) => "A${'a' * length}h!";

main() {
  final values = [1, 2, 3, 5, 10, 50];
  values.skip(1).take(3).map(scream).forEach(print);
}
```

```
Aaah!
Aaaah!
Aaaaaah!
```