Задание 1   
  
class Person {  
  String name;  
  int age;  
    
  Person( this.name, this.age);  
}   
  
void main() {  
  var person = Person("Тимофей", 33);  
  print("Имя: ${person.name}\nВозраст: ${person.age}");  
}  
  
Задание 2  
  
class Person {  
  String name;  
  int age;  
    
  Person( this.name, this.age);  
  void sayHello() {  
    print("Привет, меня зовут ${this.name}!");  
  }  
  
  
}   
  
void main() {  
  var person = Person("Тимофей", 33);  
  person.sayHello();  
}  
  
Задание 3  
  
class Person {  
  String name;  
  int age;  
    
  Person( this.name, this.age);  
  void sayHello() {  
    print("Привет, меня зовут ${this.name}!");  
  }  
  
  
}   
  
class Student extends Person {  
  String studentId;  
    
  Student(String name, int age, this.studentId) : super(name, age);  
    
  void study () {  
    print("Я студент с ID ${this.studentId} и я учусь!");  
  }  
}  
  
void main(){  
    
  var student = Student("Тимофей",33, "54321");  
    
  student.sayHello();  
  student.study();  
}  
  
Задание 4  
  
abstract class Shape {  
  double calculateArea();  
}  
class Circle extends Shape {  
  double radius;  
    
  Circle(this.radius);  
    
  @override  
  double calculateArea(){  
    return 3.14 \* radius \* radius;  
      
  }  
}  
  
class Rectangle extends Shape {  
  double width;  
  double height;  
    
  Rectangle(this.width, this.height);  
    
  @override  
  double calculateArea(){  
    return width \* height;  
      
  }  
}  
void main (){  
  var circle = Circle (5);  
  var rectangle = Rectangle (4, 6);  
    
  print (" Площадь круга: ${circle.calculateArea()}");  
  print (" Площадь прямоугольника: ${rectangle.calculateArea()}");  
}  
  
Задание 5  
  
abstract class Drawable {  
  void draw ();  
}  
  
class Circle implements Drawable {  
  @override  
  void draw() {  
    print ("Круг нарисован!");  
  }  
}  
  
void main () {  
   var circle = Circle();  
   circle.draw();  
}

Задание 6  
class MathUtils {  
  static double calculateCircleArea(double radius) {  
    return 3.14 \* radius \* radius;  
  }  
}  
  
void main() {  
  double radius = 5;  
  double area = MathUtils.calculateCircleArea(radius);  
  print("Площадь круга с радиусом $radius: $area");  
}  
  
Задание 7  
  
  class Person{  
    late int \_age;  
    
  int get age => \_age;  
    
  set age (int value) {  
    if (value >= 0) {  
      \_age = value;  
    } else {  
      print( "Возраст не может быть отрицательным!") ;  
    }  
  }  
}  
  void main() {  
var person = Person();  
person.age = 30;  
print ("Возраст: ${person.age}");  
person.age = -5;  
    //выведет сообщение об ошибке.  
  }  
  
Задание 8  
  
class Logger {  
  static final Logger \_instance = Logger.\_interal();  
  
  factory Logger () {  
    return \_instance;  
  }  
    
  Logger .\_interal();  
    
  void log(String message) {  
    print ("Лог: $message");  
  }  
}  
  
void main () {  
  var logger = Logger();  
  logger.log("ДПК");  
}  
  
Задание 9  
  
enum DayOfWeek {  
  monday.  
  tuesday.  
  wednesday.  
  thursday.  
  friday.  
  saturday.  
  sunday.  
}  
  
void checkWeekend(DayOfWeek) {  
  if (day == DayOfWeek.saturday || day == DayOfWeek.sunday) {  
    print ("Это выходной день!");  
  } else {  
    print ("Это рабочий день.");  
  }  
  }  
void main () {  
  checkWeekend(DayOfWeek.saturday);  
  checkWeekend(DayOfWeek.monday);  
}  
  
Задание 10  
mixin Flyable {  
  void fly () {   
    print ("Я могу летать!");  
  }  
}  
  
class Bird with Flyable {  
  //Дополнитльные свойства и методы класса Bird  
}  
  
void main () {  
  var bird = Bird ();  
  bird.fly( ) ;  
}

Задание 6  
class MathUtils {  
  static double calculateCircleArea(double radius) {  
    return 3.14 \* radius \* radius;  
  }  
}  
  
void main() {  
  double radius = 5;  
  double area = MathUtils.calculateCircleArea(radius);  
  print("Площадь круга с радиусом $radius: $area");  
}  
  
Задание 7  
  
  class Person{  
    late int \_age;  
    
  int get age => \_age;  
    
  set age (int value) {  
    if (value >= 0) {  
      \_age = value;  
    } else {  
      print( "Возраст не может быть отрицательным!") ;  
    }  
  }  
}  
  void main() {  
var person = Person();  
person.age = 30;  
print ("Возраст: ${person.age}");  
person.age = -5;  
    //выведет сообщение об ошибке.  
  }  
  
Задание 8  
  
class Logger {  
  static final Logger \_instance = Logger.\_interal();  
  
  factory Logger () {  
    return \_instance;  
  }  
    
  Logger .\_interal();  
    
  void log(String message) {  
    print ("Лог: $message");  
  }  
}  
  
void main () {  
  var logger = Logger();  
  logger.log("ДПК");  
}  
  
Задание 9  
  
enum DayOfWeek {  
  monday.  
  tuesday.  
  wednesday.  
  thursday.  
  friday.  
  saturday.  
  sunday.  
}  
  
void checkWeekend(DayOfWeek) {  
  if (day == DayOfWeek.saturday || day == DayOfWeek.sunday) {  
    print ("Это выходной день!");  
  } else {  
    print ("Это рабочий день.");  
  }  
  }  
void main () {  
  checkWeekend(DayOfWeek.saturday);  
  checkWeekend(DayOfWeek.monday);  
}  
  
Задание 10  
mixin Flyable {  
  void fly () {   
    print ("Я могу летать!");  
  }  
}  
  
class Bird with Flyable {  
  //Дополнитльные свойства и методы класса Bird  
}  
  
void main () {  
  var bird = Bird ();  
  bird.fly( ) ;  
}

Задание 6  
class MathUtils {  
  static double calculateCircleArea(double radius) {  
    return 3.14 \* radius \* radius;  
  }  
}  
  
void main() {  
  double radius = 5;  
  double area = MathUtils.calculateCircleArea(radius);  
  print("Площадь круга с радиусом $radius: $area");  
}  
  
Задание 7  
  
  class Person{  
    late int \_age;  
    
  int get age => \_age;  
    
  set age (int value) {  
    if (value >= 0) {  
      \_age = value;  
    } else {  
      print( "Возраст не может быть отрицательным!") ;  
    }  
  }  
}  
  void main() {  
var person = Person();  
person.age = 30;  
print ("Возраст: ${person.age}");  
person.age = -5;  
    //выведет сообщение об ошибке.  
  }  
  
Задание 8  
  
class Logger {  
  static final Logger \_instance = Logger.\_interal();  
  
  factory Logger () {  
    return \_instance;  
  }  
    
  Logger .\_interal();  
    
  void log(String message) {  
    print ("Лог: $message");  
  }  
}  
  
void main () {  
  var logger = Logger();  
  logger.log("ДПК");  
}  
  
Задание 9  
  
enum DayOfWeek {  
  monday,  
  tuesday,  
  wednesday,  
  thursday,  
  friday,  
  saturday,  
  sunday,  
}  
  
void checkWeekend(DayOfWeek) {  
  if (day == DayOfWeek.saturday || day == DayOfWeek.sunday) {  
    print ("Это выходной день!");  
  } else {  
    print ("Это рабочий день.");  
  }  
  }  
void main () {  
  checkWeekend(DayOfWeek.saturday);  
  checkWeekend(DayOfWeek.monday);  
}  
  
Задание 10  
mixin Flyable {  
  void fly () {   
    print ("Я могу летать!");  
  }  
}  
  
class Bird with Flyable {  
  //Дополнитльные свойства и методы класса Bird  
}  
  
void main () {  
  var bird = Bird ();  
  bird.fly( ) ;  
}

Задание 11

Class Animal {}

Class Dog extends Animal {}

Class Cat extends Animal {}

Void main () {

Var animals = [Dog(), Cat(), Dog(), Cat()];

For(var animal in animals) {

If(animal is Dog) {

Print(«Это собака!»);

}

Else if(animal is Cat) {

Print(«Это кошка!»);

}

}

}

Задание 12

Class Vehicle {}

Class Car extends Vehicle {}

Void main() {

Vehicle vehicle = Car();

If (vehicle is Car) {

Car car = vehicle as Car;

Print («Объект успешно приведен к типу Car.»);

} else {

Print («Обьект не является типом Car.»);

}

}

Задание 13

Class Person {

String? Name;

Person(this.name);

}

Void main() {

Var person = Person(null);

Print(«Имя: ${person.name ?? «Неизвестно»}»);

}

Задание 14

Class Address {

String city;

Address(this.city);

}

Class Person {

Address? Address;

Person(this.address);

}

Void main() {

Var person = Person(null);

Print(«Город: ${person.address?.city ?? «Неизвестно»}»);

}

Задание 15

Class Person {

String name = «»;

Int age = 0;

Void setName(String name) {

This.name = name;

}

Void setAge(int age) {

This.age = age;

}

}

Void main () {

Var person = Person()

..setName(«Тимофей»)

..setAge(33);

Print(«Имя: ${person.name}, Возраст: ${person.age}»);

}

Задание 16

Class Address {

String city;

Address(this.city);

}

Class Person {

Late Address sddress;

Void setAddress(Address address) {

This.sddress = address;

}

}

Void main() {

Var person = Person();

Person.setAddress(Address(«Донской»));

Print(«Город; ${person.sddress.city}»);

}

Задание 17

Class Person {

String name;

Int age;

Person ({required this.name, required this.age});

}

Void main() {

Var person = Person (name: «Тимофей», age: 33);

Print(«Имя: ${person.name},Возраст: ${person.age}»);

}

Задание 18

Class Point {

Final int x;

Final int y;

Const Point (this.x, this.y);

}

Void main() {

Const point = Point (10, 20);

Print («Точка☹${point.x},${point.y})»);

}

Задание 19

Class Circle {

Final double radius;

Circle (this.radius);

}

Void main( ) {

Var circle = Circle(5);

Print(«Радиус круга: ${circle.radius}»);

}

Задание 20

Class Person{

Int age;

Person(this.age) : assert(age >= 0, «Возраст на может быть отрицательным! «);

}

Void main() {

Var person1 = Person(30);

Print («Возраст: ${person1.age}»);

// Этот код вызовет ошибку, так как возраст отрицательный

// var person2 = Person(-5);

}