***Dart***

Print(“hello word”);

print("$abd $x");🡺عشان نحط اكثر من متغير في امر الطباعه

or print(abd + x);🡺اذا كانو نفس التعريف سترنغ مع سترنغ

string? Name🡺هيك ممكن تكون قيمه المتغير null

int x=10; int a=5;

int c=x~/a;🡺~يجب وجود هذه الاشارة حتى يطلع الجواب انتجر

الجمل الشرطيه نفس باقي لغات البرمجه

If(abdalla==hijjawe)

Print(“true”)

Else

Print(“false”)

switch(price){

case 1000:print("ok");

default: print("no"); break; }

If(abdalla==hijjawe)

Print(“true”)

Else if(abdalla==kaboos)

Print(“true”)

Else

Print(“false”)

    for(int i=0;i<100;i+=2){

        print(i); }🡺الشكل العام للفور

int x=10;

   while(x<100){

    print(x);

    x++;}

int x=10;

   do{

    print(x);

    x++;

   }while(x<100);

   List abdalla=["kaboos","hijjawe","motasem"];

مثال ع القوائم

   abdalla.forEach((abdalla) { print(abdalla);});

يطبع القائمه بشكل اللوب وتسمح باختلاف العدد

List abdalla=["kaboos","hijjawe","motasem"];

   abdalla.addAll(["jenana","omar"]);

اضافه عناصر القائمة الثانيه لعناصر القائمه الاولى

   abdalla.insert(0,"omar");

اضافه العنصر الى القائمه الاولى مع تحديد المكان

Insertall🡺نفس العاديه بس هاي مخصصه لاضافه القوائم

Abdalla.replacerange(0,2,[“1”,”2”])🡺 هيك رح يشيل الماضيات ويحط الجداد حسب الترتيب يلي انت حاطه

   Map kaboos={"name":"abdlla","age":"19" };

كل كلمه في الها مفتاح

ممكن نحكي لما نطبع

Print(kaboos)OR print(kaboos[“name”])

Set name ={"abdalla","hijjawe"};

ميزه السيت هي عدم التكرار اي لا تستطيع تكرير عبدالله

Print(name.tolist());🡺 تحويل السيت للست

List name=[];

Map info={“name”:”abdalla”,”age”:22};

Info.foreach((key,value){name.add(value);})

Print(name);🡺abdalla 22

لو في عنا ملف وبدنا نستدعيه بملف الرئيسي

Import’namefile.dart’

اذا كان الملف بمجلد اخر

Import’namefolder/namefile’

اذا كان الملف خارج المجلد الرئيسي

Import’../namefile.dart’

void main()

{

List abdalla=[10,20,30,40,50,60];

double avg=0;

for(int x=0;x<abdalla.length;x++){

  avg +=abdalla[x];

}

double avg1=avg/abdalla.length;

print(avg1);

}

برنامج حساب الافريج

Import ‘namefile.drat’ as abdalla🡺هيك بنخلي المكتبه باسم عبدالله عشان نطبق الاكواد يلي موجوده جواتها لازم نعمل

Abdalla.print

String name="abdalla";

print(name.codeUnits);🡺 [97, 98, 100, 97, 108, 108, 97]

بعمل ترميز للاسم

Ex for runs:

String name =”abdalla”

For(int i=0;i<name.lentgh;i++){

Print(name[i])}🡺رح يطبع عبدالله حرف حرف

assert(a!=10,"a=10");🡺

اذا الف تساوي عشرة رح يعطي ايرور

List<string>name🡺هيك ما رح تستقبل اللست غير قيم السترنغ بس

List name=["abdalla","hijjawe","kaboos","omar","king"];

  List sublist=name.sublist(1,3);

هيك رح يطبع خانه واحد وخانه اثنين

Name.shuffle🡺بتغير ترتيب اللست بشكل عشوائي

Map namespam=name.asmap🡺تحويل اللست لماب

Name.contens(a)🡺هل الاسم يحتوي ع حرف الالف

Name.startwith(a)🡺هل اول حرف من الاسم الف

Name.endwith(a)🡺هل اخر الاسم يحتوي ع الالف

Name.indexof(a)🡺هل يحتوي الاسم على حرف الالف وكم رقم خانته

List name=[“abdalla”,”hijjawe”,”ahmed”]

List namestart=name.where((a)=>a.startwith(“a”).tplist()

🡺[abdalla,ahmed]

**Iterable🡺يلي بقدر اعمل عليه تكرار**

Ex :for(string namevarible in namelist){

Print name}

# Iterator 🡺معناها تكرار

List name=["abdalla","hijjawe","kaboos","omar","king"];

  Iterator iter=name.iterator;

  while(iter.moveNext()){

    print(iter.current);

# النتيجه رح يطبع الاسماء باللست ع الترتيب

# ممكن نستخدم بدل لست iterable

# الست والليست نوع من انواع الاتريبل

Map info={"name":"abdalla","age":19,"phone":123456};

 Iterable infoabd = info.entries.map((e) => e.key);

 print(infoabd);

# result= (name, age, phone)

 int a=10;

  int b=0;

try{

  int res=a~/b;

  print(res);

}catch(anyname){

  print("error : $anyname");

}

# Ex

**OOP**

void main()

{Abdalla hijjawe = new Abdalla();

 print(hijjawe.screen);}

class Abdalla {String screen="6.4";

  String name="iphone";}

طريقه استخدام الكلاس في الفنكشين الرئيسي

Abdalla hijjawe = new Abdalla("kaboos");

}

class Abdalla {

 Abdalla(screen){

print(screen);

# مثال ع Constructor

void main()

{Abdalla hijjawe = new Abdalla()..name=”kaboos”;

 print(hijjawe.screen);}

class Abdalla {String screen="6.4";

  String name="iphone";}

# طريقه ثانيه لتعين قيمه

# Get🡺يقدر يجبلك قيمه من داخل الكلاس

# Set🡺وظيفتها اعادة تعريف القيم

new Abdalla().kaboos();

طريقه استدعاء سريعه للكلاس

او بحال كان بدك اكثر من خيار

New Abdalla()..kaboos()..joker();

void main()

{

empo abdalla=new empo();

abdalla.postion="cook";

abdalla.salary="2000";

abdalla.phone="0779550604";

abdalla.name;

}

class Job {

String name="cyber";

String salary="1000";

String postion="manger";

}

class empo extends Job{

String phone="077";

}

مثال ع الوراثه inheritance

  return super.name;

روح للكلاس الاصلي يلي انعمل منه وراثه

اذا الكلاس الثالث عامل وراثه من الكلاس الثاني والثاني فيه كونستركتر يجب ع الثالث ان يكون فيه كونستركتر بهذا الطريقه

Name class(this.username):super(username);

emp abdalla=new emp();

 abdalla.salary;

 abdalla.name;

}

abstract class Job{

  String name="cyber";

  String salary="1000";}

class emp extends Job{

  String postion="manger";}

لكن تقدر تعمله وراثهAbstract ما بتقدر تستدعي كلاس مبدوء ب

# implements class🡺بتخلي كلاس الابن يورث جميع متغيرات كلاس الاب

# ex:class abdalla{

# int name;}

# class kaboos implements abdalla{

# int name🡺اجباري}

void main()

{

print(phone.values);

}

enum phone{

  samsung,iphone,tchaomi,huawei

}

مثال ع enum

# regular expressions🡺مهمتها التحقق من المدخلات

String patternEmail=r"^[a-zA-Z0-9.a-zA-Z0-9.!#$%&'\*+-/=?^\_`{|}~][+@[a-zA-Z0-9]+\.[a-zA-Z](mailto:+@[a-zA-Z0-9]+\.%5ba-zA-Z)]+";==> من الانترنت بتنجاب

RegExp abdalla=new RegExp(patternEmail);

String email="abdalla@gmail.com";

print(abdalla.hasMatch(email));

# التاكد من الايميل اذا هو صح او لا

# \_abdalla🡺هيك ما رح تقدر توخذ المتغيربصفحه ثانيه لانه صار برايفت

تكمله للبرايفت مراجعه فيديو69

Class three extends one with two🡺وراثه كلاس واحد واثنين لثلاث

mixin abdalla{

  String kaboos="abdalla";

  String phone="077";

}

class joker  with abdalla{

  String name="king";

}

مثال ع المكسين وبامكان نطبق القاعده يلي فوق

**The End**