

TP 05 – Le langage Dart

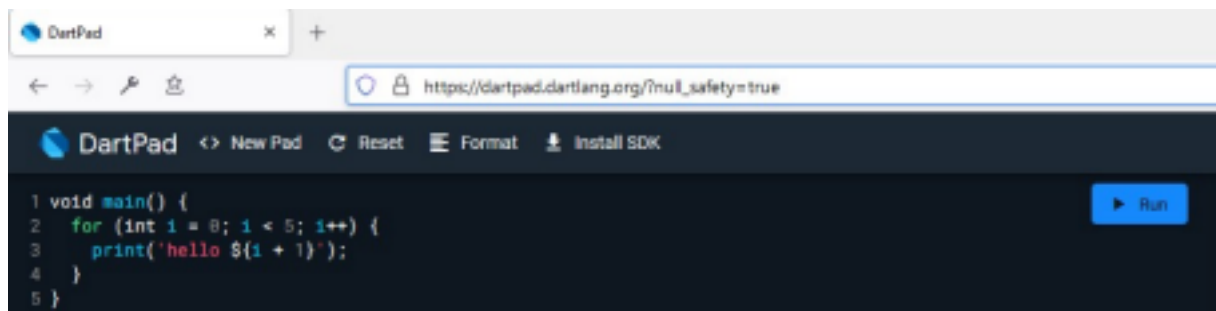
Objectifs :

Maîtriser le langage Dart et ces concepts de base.

Étapes à suivre :

1. Allez vers l'adresse Web suivante : <https://dartpad.dartlang.org>

Par défaut vous allez trouver le contenu suivant :



Si vous appuyer sur le bouton (Run), vous allez avoir le résultat suivant :

<pre>void main() {   for (int i = 0; i &lt; 5; i++) {     print('hello \${i + 1}');   } }</pre>	<pre>hello 1 hello 2 hello 3 hello 4 hello 5</pre>
---	--

2. Écrire le code suivant puis lancez l'exécution :

<pre>void main() {   print('hello'); }</pre>	<pre>hello</pre>
--	------------------

3. Déclarez une variable de type entier « int » appelée « age » et lui affectez la valeur « 30 » :

<pre>void main() {   int age = 30;   print(age); }</pre>	<pre>30</pre>
--	---------------

4. Déclarez une variable appelée « nom » de type chaîne de caractère « String » et lui affectez la valeur « KBR » :

<pre>void main() {   String nom = 'KBR';   print(nom); }</pre>	<pre>KBR</pre>
--	----------------

5. Changez la valeur de la variable « nom » après sa déclaration pour avoir comme valeur « BEN » :

<pre>void main() {   String nom = 'KBR';   nom = 'BEN';   print(nom); }</pre>	<pre>BEN</pre>
---	----------------

6. Déclarez une variable appelée « estActif » de type booléen « bool » et lui affectez la valeur « false » :

<pre>void main() {   bool estActif = false;   print(estActif); }</pre>	<pre>false</pre>
--	------------------

7. Déclarez une variable appelée « nom » de type dynamique « dynamic » et lui affectez la valeur « KBR », puis lui affecter la valeur « 30 » :

<pre>void main() {   dynamic nom = 'KBR';   nom = 30;   print(nom); }</pre>	<pre>30</pre>
---	---------------

8. Vérifiez le retour de la fonction « main » :

<pre>void main() {   print('une chose');   return 'bonjour'; }</pre>	<pre>Error compiling to JavaScript: Info: Compiling with sound null safety Warning: Interpreting this as package URI, 'package:dartpad_sample/main.dart' lib/main.dart:3:9: Error: Can't return a value from a void function.   return 'bonjour';   ^ Error: Compilation failed.</pre>
--	--

9. Déclarez une fonction appelée « salutation » qui retourne une chaîne de caractères « bonjour », puis déclarez une fonction appelée « monAge » qui retourne un entier, puis lui affecter la valeur « 30 » :

<pre>void main() {     print('une chose');     String salut = salutation();     int age = monAge();     print(salut);     print(age); } String salutation(){     return 'bonjour'; } int monAge(){     return 30; }</pre>	<pre>une chose bonjour 30</pre>
---	---

10. Changez la déclaration des fonctions « salutation » et « monAge » comme suit :

<pre>void main() {     print('une chose');     String salut = salutation();      int age = monAge();     print(salut);     print(age); }</pre>	
<pre>String salutation() =&gt; 'bonjour'; int monAge() =&gt; 30;</pre>	<pre>une chose bonjour 30</pre>

11. Déclarez un tableau appelé « noms » de type « List » et lui affecter un ensemble de chaînes de caractères :

<pre>void main() {     List noms =     ['mohamed', 'zainebe', 'taha'];     print(noms); }</pre>	<pre>[mohamed, zainebe, taha]</pre>
---	-------------------------------------

12. Ajoutez un élément à la fin du tableau de valeur « ali » :

<pre>void main() {   List noms = ['mohamed','zaineB','taha']; noms.add('ali'); print(noms); }</pre>	<pre>[mohamed, zaineB, taha, Ali]</pre>
---	---

13. Supprimez un élément du tableau :

<pre>void main() {   List noms = ['mohamed','zaineB','taha']; noms.add('ali'); noms.remove('taha'); print(noms); }</pre>	<pre>[mohamed, zaineB, Ali]</pre>
--	-----------------------------------

14. Ajoutez un entier de valeur « 30 » au tableau :

<pre>void main() {   List noms = ['mohamed','zaineB','taha']; noms.add('ali'); noms.remove('taha'); noms.add(30); print(noms); }</pre>	<pre>[mohamed, zaineB, Ali, 30]</pre>
--	---------------------------------------

15. Changez le tableau noms de manière qu'il ne sauvegarde que des chaînes de caractères :

<pre>void main() {   List&lt;String&gt; noms = ['mohamed','zaineB','taha']; noms.add('ali'); noms.remove('taha'); noms.add(30); print(noms); }</pre>	<pre>Error compiling to JavaScript: Info: Compiling with sound null safety Warning: Interpreting this as package URI, 'package:dartpad_sample/main.dart '. lib/main.dart:6:12: Error: The argument type 'int' can't be assigned to the parameter type 'String'.   noms.add(30);               ^ Error: Compilation failed.</pre>
--	--

16. Créez une classe appelée « Utilisateur » qui déclare deux attributs: nom (chaîne de caractères) et age (entier), puis déclare une fonction appelée « login() » qui retourne void :

<pre> void main() {     Utilisateur premierUtilisateur =     Utilisateur();     print(premierUtilisateur.nom);     print(premierUtilisateur.age);     premierUtilisateur.login(); } class Utilisateur {     String nom = 'BEN';     int age = 40;      void login(){         print('utilisateur connecté');     } } </pre>	<pre> BEN 40 utilisateur connecté </pre>
--	--

17. Créez un deuxième objet de type « Utilisateur » comme suit, puis commentez le résultat:

<pre> void main() {     Utilisateur premierUtilisateur =     Utilisateur();     print(premierUtilisateur.nom);     print(premierUtilisateur.age);     premierUtilisateur.login();     Utilisateur deuxiemeUtilisateur =     Utilisateur();     print(premierUtilisateur.nom); } class Utilisateur {     String nom = 'BEN';     int age = 40;      void login(){         print('utilisateur connecté');     } } </pre>	<pre> BEN 40 utilisateur BEN </pre>
--	-------------------------------------

18. Créez un constructeur au niveau de la classe « Utilisateur » :

<pre> void main() {     Utilisateur premierUtilisateur =     Utilisateur('KBR', 40);     print(premierUtilisateur.nom); } class Utilisateur {     String nom="";     int age=0;      Utilisateur(String nom, int age){         this.nom = nom;         this.age = age;     }      void login(){ </pre>	<pre> KBR </pre>
--	------------------

<pre>         print('utilisateur connecté');     } } </pre>	
---	--

19. Testez l'exécution en appelant deux utilisateurs :

<pre> void main() {     Utilisateur premierUtilisateur =     Utilisateur('KBR', 40);     print(premierUtilisateur.nom);     Utilisateur deuxiemeUtilisateur =     Utilisateur('KBR2', 30);     print(deuxiemeUtilisateur.nom); } class Utilisateur {     String nom="";     int age=0;      Utilisateur(String nom, int age){         this.nom = nom;         this.age = age;     }      void login(){         print('utilisateur connecté');     } } </pre>	<pre> KBR KBR2 </pre>
--	-----------------------

20. Ajoutez une nouvelle classe appelée « SuperUtilisateur » qui étend la classe « Utilisateur » et qui implémente la fonction « publier() » et en héritant du constructeur de la classe parente « Utilisateur » :

<pre> void main() {     Utilisateur premierUtilisateur =     Utilisateur("KBR", 40);     print(premierUtilisateur.nom);     Utilisateur deuxiemeUtilisateur =     Utilisateur("KBR2", 30);     print(deuxiemeUtilisateur.nom); } class Utilisateur {     String nom="";     int age=0;      Utilisateur(String nom, int age){         this.nom = nom;         this.age = age;     }      void login(){         print('utilisateur connecté');     } } </pre>	<div>KBR</div> <div>KBR2</div>
<pre> class SuperUtilisateur extends Utilisateur { SuperUtilisateur(String nom, int age) : super(nom, age); void publier(){ print('publication'); } } </pre>	

21. Ajoutez une référence à la classe « SuperUtilisateur » et appelez la fonction « publier() » :

<pre> void main() {     Utilisateur premierUtilisateur =     Utilisateur("KBR", 40);     print(premierUtilisateur.nom);     Utilisateur deuxiemeUtilisateur =     Utilisateur("KBR2", 30);     print(deuxiemeUtilisateur.nom);     SuperUtilisateur     troisiemeUtilisateur =     SuperUtilisateur('Super KBR', 20);     print(troisiemeUtilisateur.nom);     troisiemeUtilisateur.publier();  } class Utilisateur {     String nom="";     int age=0;      Utilisateur(String nom, int age){         this.nom = nom;         this.age = age;     }      void login(){         print('utilisateur connecté');     } } class SuperUtilisateur extends Utilisateur {     SuperUtilisateur(String nom, int age) : super(nom, age);      void publier(){         print('publication');     } } </pre>	<pre> KBR KBR2 Super KBR publication </pre>
--	---

22. Testez l'appel de la fonction « publier() » à partir de la variable « deuxiemeUtilisateur » de type « Utilisateur » :

<pre> void main() {     Utilisateur premierUtilisateur =     Utilisateur("KBR", 40);     print(premierUtilisateur.nom); </pre>	<pre> Console </pre>
--	----------------------



<pre> Utilisateur deuxiemeUtilisateur = Utilisateur('KBR2', 30); print(deuxiemeUtilisateur.nom); SuperUtilisateur troisiemeUtilisateur = SuperUtilisateur('Super KBR', 20); print(troisiemeUtilisateur.nom); troisiemeUtilisateur.publier();  deuxiemeUtilisateur.publier();  } class Utilisateur {   String nom="";   int age=0;    Utilisateur(String nom, int age){     this.nom = nom;     this.age = age;   }    void login(){     print('utilisateur connecté');   } } class SuperUtilisateur extends Utilisateur {   SuperUtilisateur(String nom, int age) : super(nom, age);    void publier(){     print('publication');   } } </pre>	<pre> Error compiling to JavaScript: Info: Compiling with sound null safety Warning: Interpreting this as package URI, 'package:dartpad_sample/main.dart' . lib/main.dart:11:23: Error: The method 'publier' isn't defined for the class 'Utilisateur'. - 'Utilisateur' is from 'package:dartpad_sample/main.dart' ('lib/main.dart').   deuxiemeUtilisateur.publier();                         ^^^^^^^ Error: Compilation failed. </pre>
--	--

23. Testez l'appel de la fonction « login() » pour l'objet « troisiemeUtilisateur » :

<pre> void main() {   Utilisateur premierUtilisateur =   Utilisateur('KBR', 40);   print(premierUtilisateur.nom);   Utilisateur deuxiemeUtilisateur =   Utilisateur('KBR2', 30);   print(deuxiemeUtilisateur.nom);   SuperUtilisateur   troisiemeUtilisateur =   SuperUtilisateur('Super KBR', 20);   print(troisiemeUtilisateur.nom);   troisiemeUtilisateur.publier();   troisiemeUtilisateur.login();  } class Utilisateur { </pre>	<pre> KBR KBR2 Super KBR publicati utilisateur connecté </pre>
--	--

<pre>String nom=""; int age=0;</pre>	
<pre>Utilisateur(String nom, int age){     this.nom = nom;     this.age = age; }  void login(){     print('utilisateur connecté'); } } class SuperUtilisateur extends Utilisateur {     SuperUtilisateur(String nom, int age) : super(nom, age);      void publier(){         print('publication');     } }</pre>	