



# Angular 2+

## Développer des applications Web

Support de cours

Réf. T44A-046



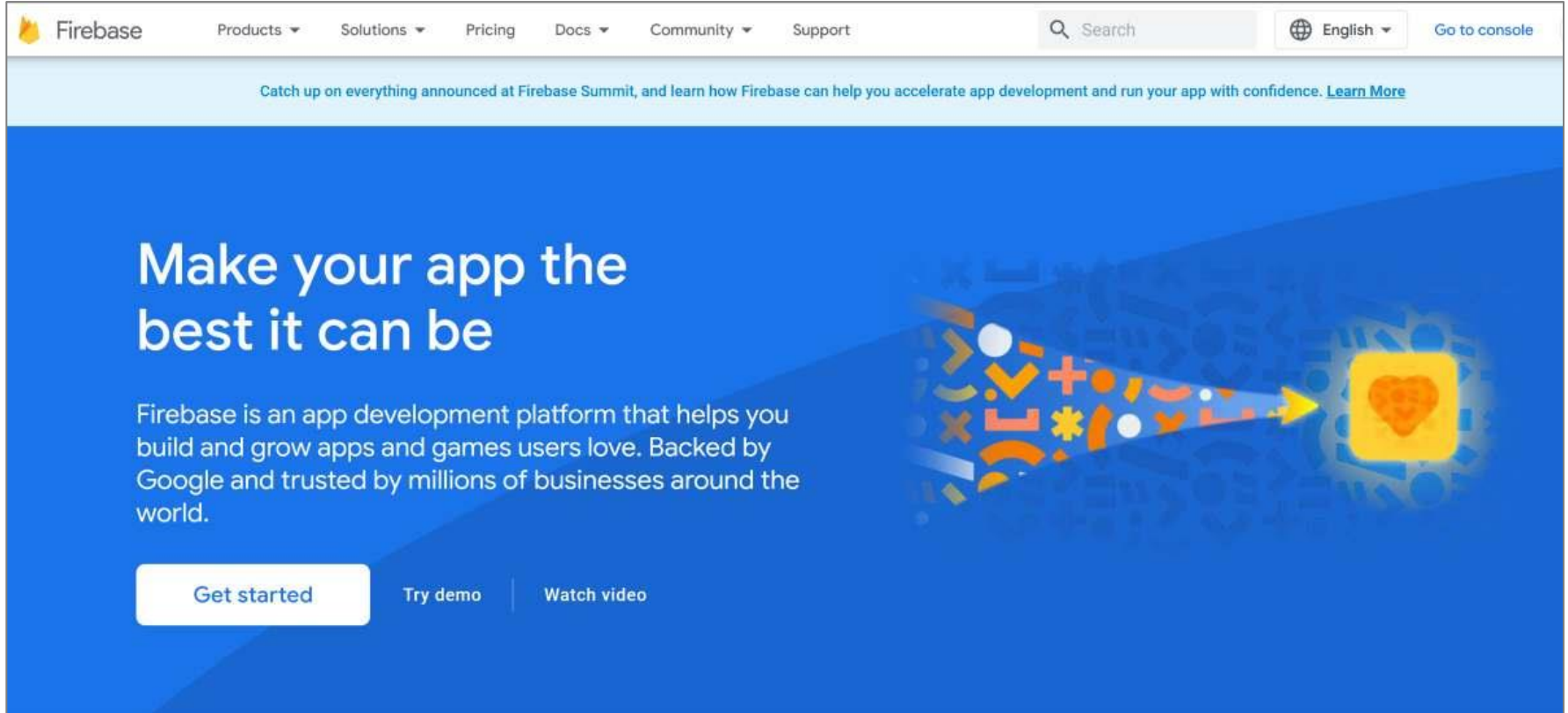


## Module 9

# Stocker ses données dans le Cloud avec RealTime de FireBase

# Contenu du module

- Mise en place d'une base de données RealTime sur FireBase
- Enregistrer les données de notre application
- Lecture des données stockées sur RealTime
- Allez plus loin avec les méthodes POST et DELETE



# Firebase : Pricing

Gratuit



## Realtime Database

Simultaneous connections ?

100

200k/database

GB stored

1 GB

\$5/GB

GB downloaded

10 GB/month

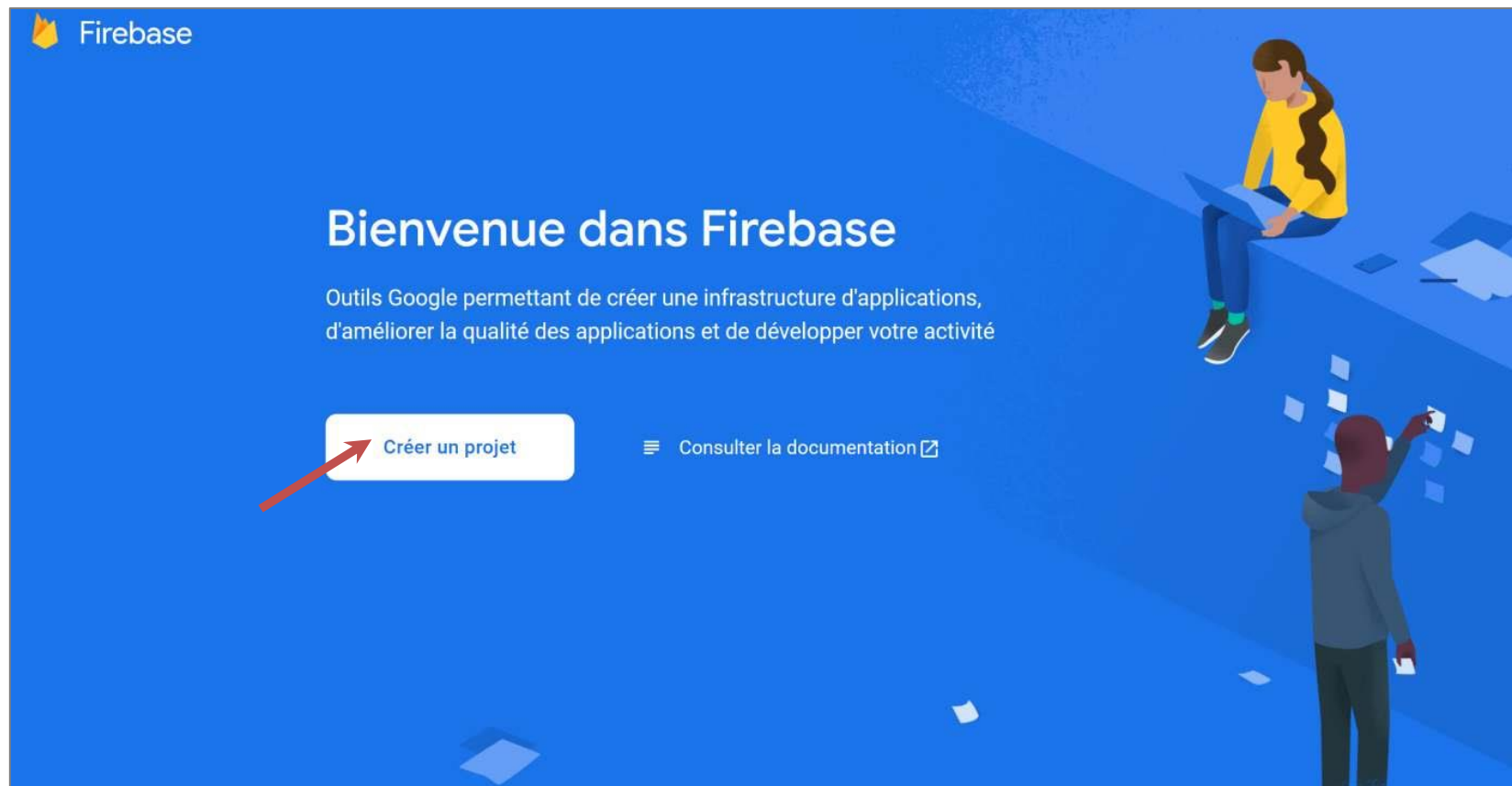
\$1/GB

Multiple databases per project

✗

✓


# FireBase : Créer un projet



# FireBase : Créer un projet

× Créer un projet(Étape 1 sur 3)

Commençons par donner un nom à votre projet<sup>?</sup>

 Saisissez le nom de votre projet

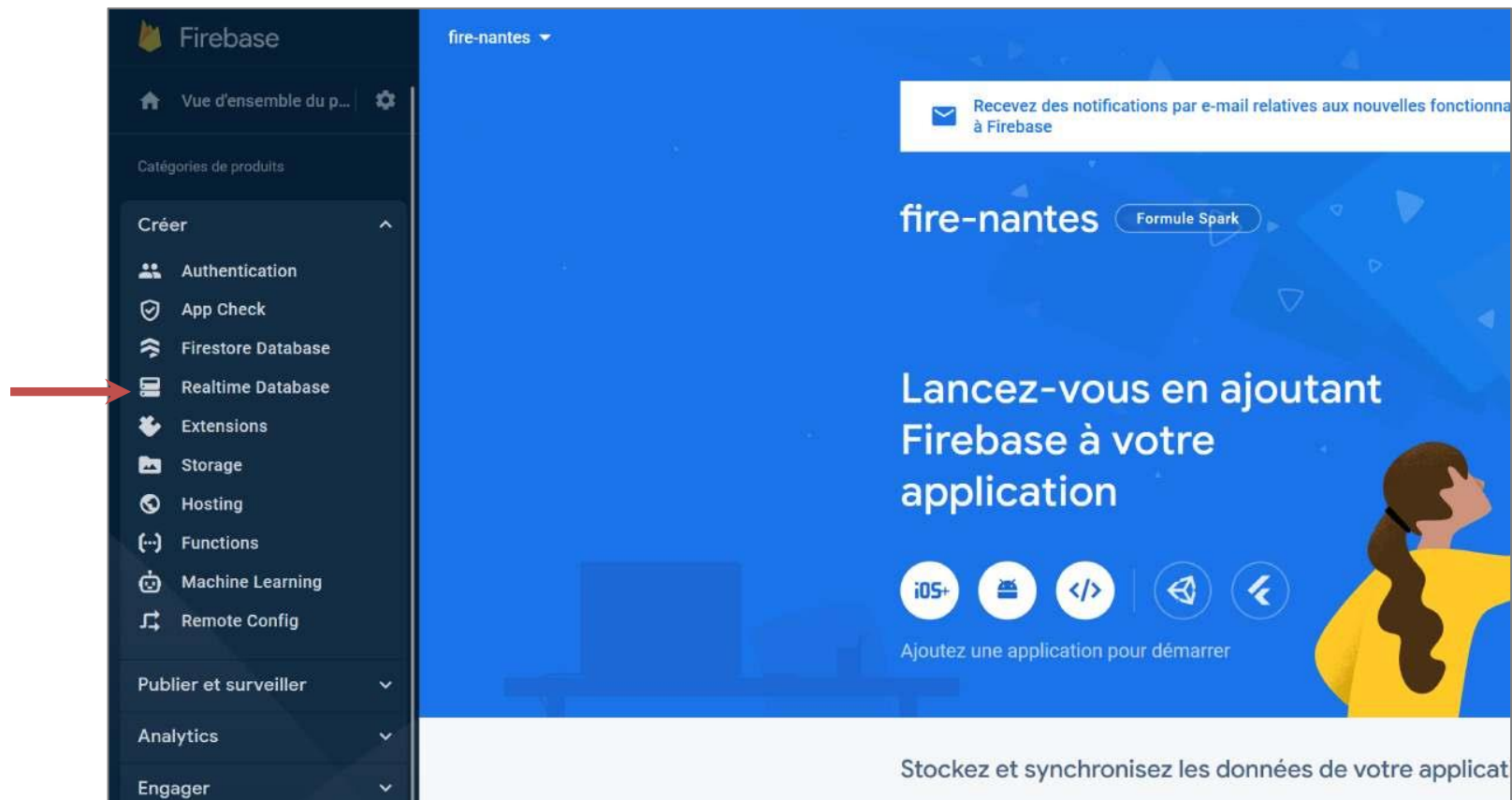
my-awesome-project-id

☐ J'accepte les [Conditions d'utilisation de Firebase](#)

☐ Je confirme que je n'utiliserai Firebase que pour mes activités commerciales, mon entreprise, mes créations ou ma profession.

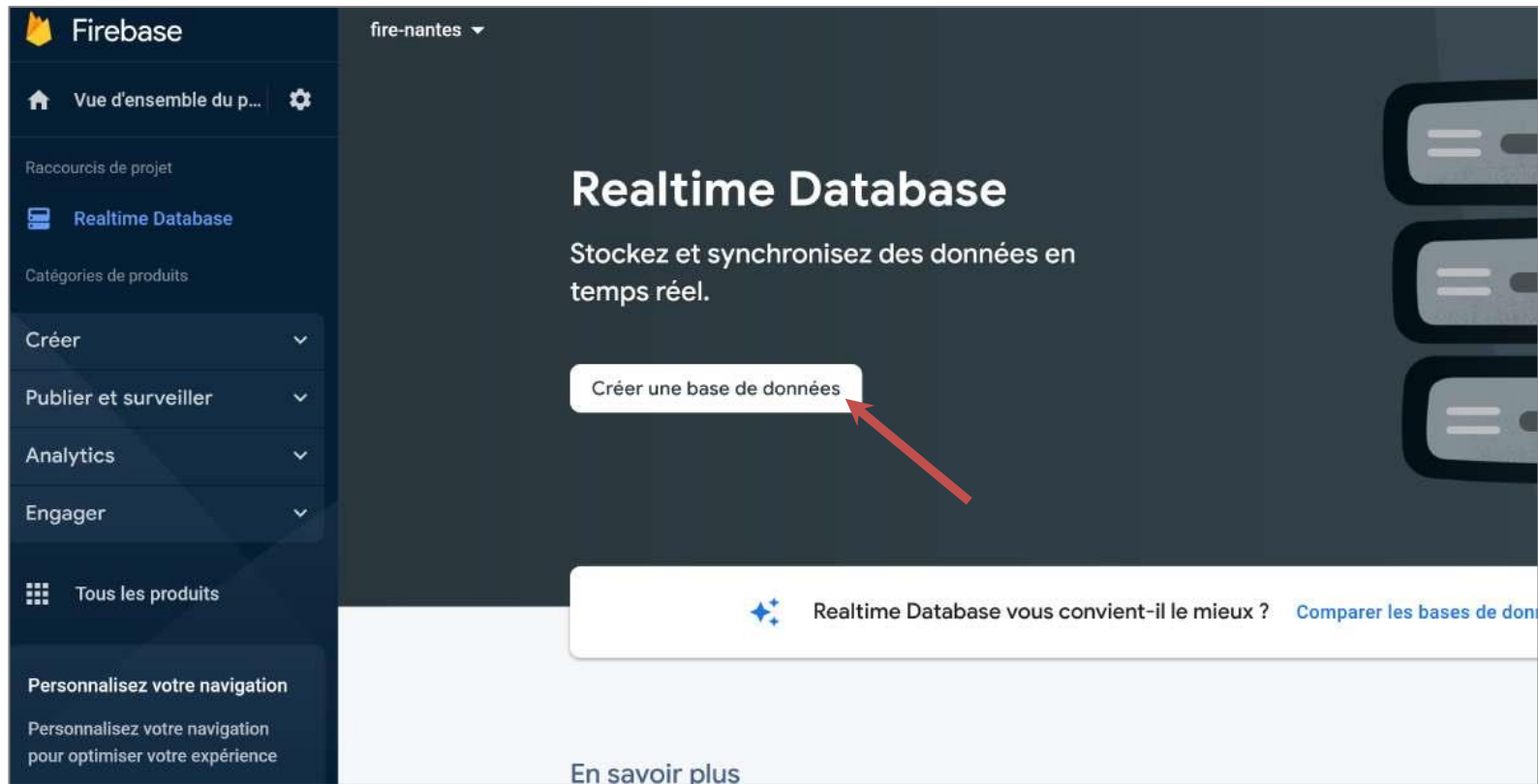
Continuer

# FireBase : Créer une base de données Real Time





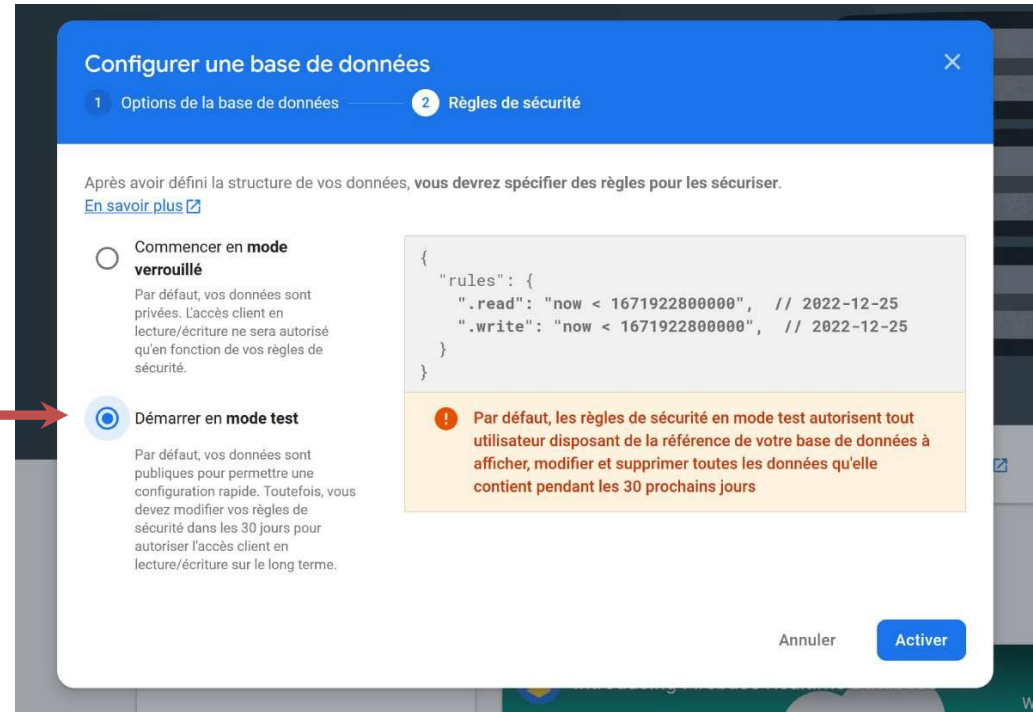
# FireBase : Créer une base de données Real Time



The screenshot shows the Firebase console interface. On the left is a dark sidebar with the 'Firebase' logo and navigation options: 'Vue d'ensemble du p...', 'Raccourcis de projet' (with 'Realtime Database' highlighted), 'Catégories de produits', and a list of product categories ('Créer', 'Publier et surveiller', 'Analytics', 'Engager') each with a dropdown arrow. At the bottom of the sidebar is 'Tous les produits' and a section for 'Personnalisez votre navigation'. The main content area has a dark header with 'fire-nantes' and a dropdown arrow. Below this, the title 'Realtime Database' is displayed in large white text, followed by the subtitle 'Stockez et synchronisez des données en temps réel.' A prominent white button with the text 'Créer une base de données' is centered. A red arrow points from the bottom right towards this button. At the bottom of the main area, there is a light blue banner with a star icon, the text 'Realtime Database vous convient-il le mieux ?' and a link 'Comparer les bases de don'. A link 'En savoir plus' is located at the very bottom center of the page.

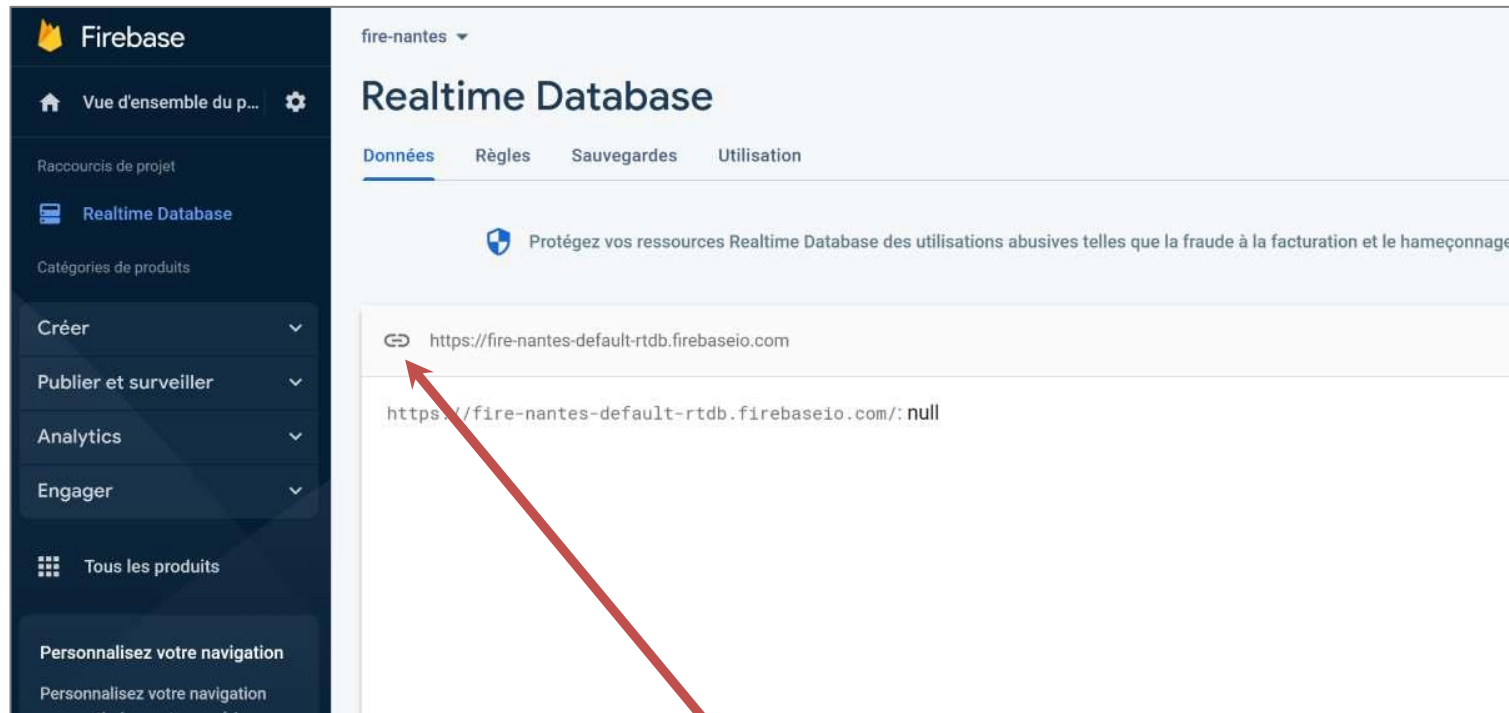
# FireBase : Choisir Mode test

- On Enlève l'authentification
  - Attention tout le monde peut lire, écrire , effacer ....
  - Penser à remettre false à la fin du TP



# FireBase : Choisir Mode test

- Copier l'URL et c'est fini ! **Backend = NO Backend**



# Enregistrer les données de notre applications

- Recopier l'URL dans le fichier TypeScript

```
export class AppComponent {  
  url = ' https://fire-nantes-default-rtdb.firebaseio.com/ ;  
}
```

- Créer un nœud ( équivalent d'une table en SQL) **fruits.json**

```
export class AppComponent {  
  url = ' https://fire-nantes-default-rtdb.firebaseio.com/fruits.json ;  
}
```



# Enregistrer les données de notre applications

- Créer une méthode PUT pour remplacer le nœud **fruits.json**

app.component.ts

```
export class AppComponent {  
  titre='';  
  fruits:string[]=[];  
  url = ' https://fire-nantes-default-rtdb.firebaseio.com/fruits.json';  
  constructor(private httpClient:HttpClient){}  
  onAjouter(){  
    this.fruits.push(this.titre);  
    this.titre = '';  
    this.httpClient.put(this.url,this.fruits).subscribe(  
      (response)=>{  
        console.log('Ecriture OK !')  
      }  
    )  
  }  
}
```

app.component.html

```
<input [(ngModel)]="titre">  
  
<button (click)="onAjouter()">  
  GO  
</button>
```

# Vérification sur FireBase

- Une ligne est ajouté , indice : 0

 <https://fire-nantes-default-rtdb.firebaseio.com>

<https://fire-nantes-default-rtdb.firebaseio.com/>

▼ — fruits

└─ 0: "pomme"

# Lecture des données stockées sur RealTime

- Dans la méthode **ngOnInit()** on vient lire avec la méthode **get** les données de FireBase

app.component.ts

```
export class AppComponent implements OnInit{
  titre='';
  fruits:string[]=[];
  url = ' https://fire-nantes-default-rtdb.firebaseio.com/fruits.json';
  constructor(private httpClient:HttpClient){}
  ngOnInit(): void {
    this.httpClient.get<string[]>(this.url).subscribe(
      (response)=>{
        if (response != undefined)
          this.fruits = response;
      }
    )
  }
}
```



## [Partie Optionnelle] BONUS

# Utiliser les méthodes POST et DELETE