### Raphaël CHRISTIAN

### Concepteur développeur d’applications

### 20/10/21

### La description du métier

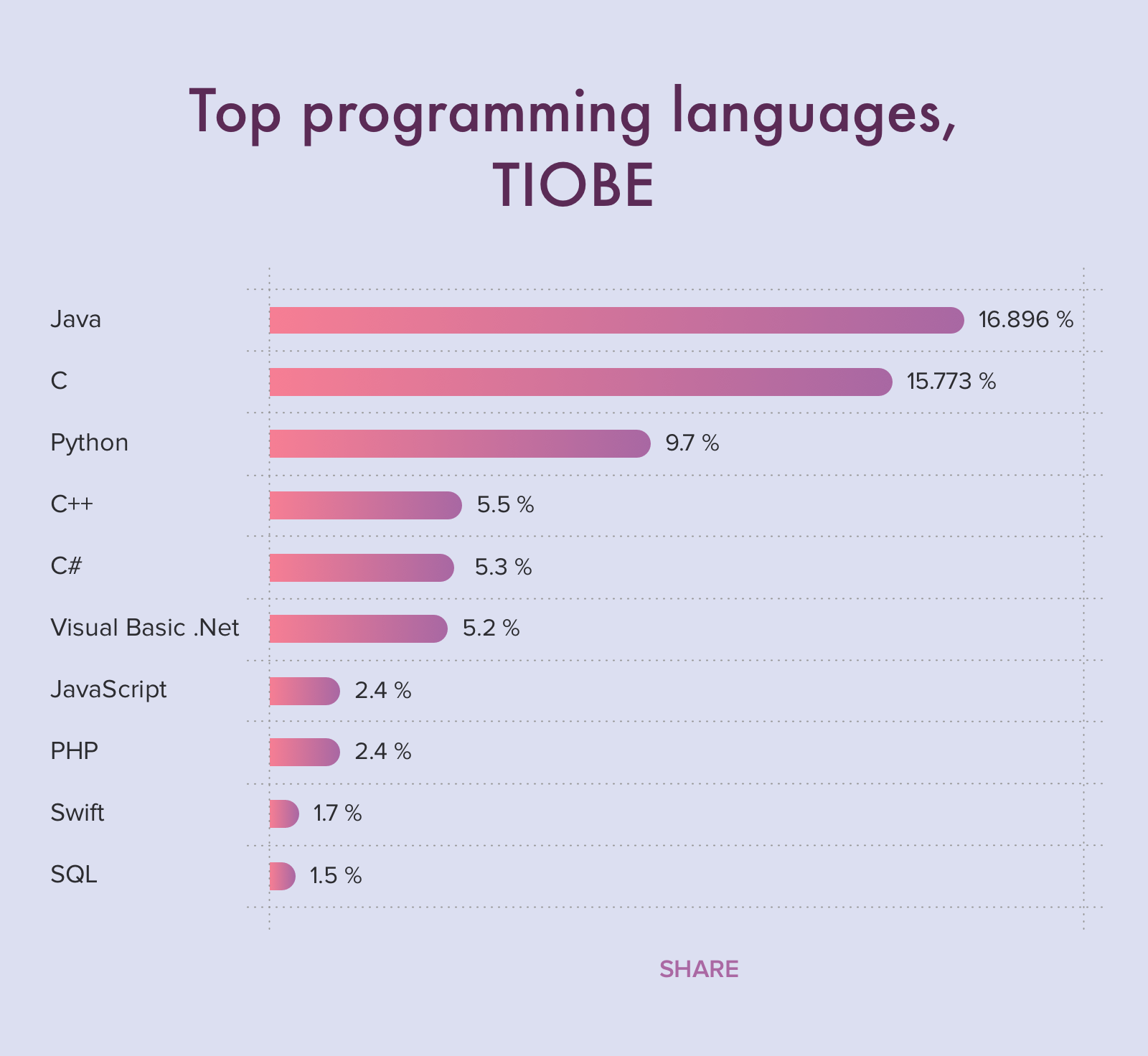
A l’issue de cette formation j’ai pour objectif d’acquérir les compétences mentionnées ci-dessous :

* Développer la partie front-end d’une interface web
* Développer la partie back-end d’une interface web
* Développer une interface desktop
* Maquetter une application
* Concevoir et mettre en place une base de données
* Concevoir et développer une application
* Collaborer autour d’un projet

### Les compétences requises sont :

* Rigueur et méthode
* Communication
* Orientation client
* Esprit d'initiative, autonomie et curiosité
* Maitrise du français et de l’anglais

### Les technologies du moment

Les langages actuels qui sont très tendance sont :

***SQL***

SQL, ou langage de requête structuré, a été ajouté à notre liste uniquement à cause de ses cas d'utilisation distincts. Bien qu'il soit assez différent des autres langages de programmation de la liste - en ce sens qu'il ne peut être utilisé seul pour un site Web ou pour le développement d'applications - nous pensons qu'il appartient ici à la fois à sa popularité et à son utilisation unique cas.

Dans le monde moderne, *les données sont essentielles*. Si vous et votre entreprise collectez et analysez les données de manière efficace, les grandes décisions seront beaucoup plus faciles à prendre. Une analyse efficace des données peut éclairer les décisions en matière de marketing, d’exploitation et de stratégie. SQL demeure le moyen le plus populaire de stockage et d’analyse de ces données.

Cela signifie que SQL est sans aucun doute le meilleur langage de programmation à apprendre si vous voulez construire une carrière. dans l'analyse des données de consommation. Les bons programmeurs SQL seront toujours en demande, la disponibilité du travail ne devrait donc pas être un problème. Sinon, SQL peut ajouter une autre chaîne à votre arc s'il travaille déjà en tant qu'ingénieur front-end. Cela vous rendra plus apte au travail, augmentera le nombre de choses que vous pouvez faire et son apprentissage devrait déboucher sur de meilleures perspectives de carrière.

Si vous aimez le son de SQL et pensez vouloir l’apprendre, consultez [**le didacticiel Interactive SQL**](https://www.bitdegree.org/course/sql-course) sur BitDegree. En enseignant les bases du codage SQL, de l’analyse de bases de données et de la rédaction de requêtes, vous n’avez pas besoin d’expérience en codage lorsque vous commencez ce cours, ce qui le rend parfait pour les débutants.

***Swift***

Bien que sorti il ​​y a seulement quatre ans, Swift est rapidement devenu l'un des langages de programmation les plus demandés en 2021. Il a été créé par Apple pour remplacer Objective-C et simplifier la programmation iOS. Il a été rapidement adopté par les programmeurs Apple et a connu une augmentation constante du classement des langues depuis.

Swift est devenu l'un des langages de programmation dont la croissance est la plus rapide. Bien que sorti en 2014, il est maintenant le langage de choix pour diverses utilisations de la programmation iOS, Mac, Apple Watch et même Web. Il est conçu pour permettre aux programmeurs d’écrire rapidement un logiciel rapide, stable et sécurisé.

Swift se base sur trois principes clés:

* *Sécurité - Swift repose sur la sécurité et la sûreté. Cela peut parfois sembler un peu restrictif, mais cela réduit le risque d'erreurs potentielles graves du développeur.*
* Vitesse - En remplacement de Objective-C et d'autres langages basés sur C, Swift doit être comparable en termes de performances. Il a donc été conçu pour être rapide, prévisible et cohérent.
* Expressivité - Contrairement aux langages plus anciens qui peuvent être contre-intuitifs et difficiles à suivre, Swift propose une syntaxe claire et prévisible, amusante et facile à utiliser.

Comme vous pouvez le constater, Swift est vraiment un langage moderne qui a beaucoup de potentiel pour l’avenir. Si vous souhaitez tirer parti de la demande de développeurs Swift, consultez [**le didacticiel Swift pour débutants**](https://fr.bitdegree.org/cours/creer-une-application-mobile) et le cours **[Learn Swift Programming](https://fr.bitdegree.org/cours/langage-swift)** sur BitDegree.

***Go***

Go, qui est un langage de programmation créé par Google en 2009 comme alternative au C et C++, est sans doute le meilleur langage de programmation à apprendre en 2021. Également connu sous le nom de GoLang, il figure désormais parmi les meilleurs langages de programmation à son extrême polyvalence.

Conçu pour être plus simple, plus sûr et plus facile à écrire, Go a pris d'assaut le monde de la programmation depuis sa sortie. Il est relativement facile à apprendre pour les débutants - une rareté parmi les langages statiquement typés - et offre une syntaxe de code très compréhensible.

En tant que langage moderne, Go offre certains avantages qui le rendent assez favorable par rapport aux langages plus anciens tels que Java et Python. Ceux-ci inclus:

* Sa vitesse. En ce qui concerne la vitesse, Go est comparable aux langages C. Cela le rend très utile pour des choses comme le développement de sites Web.
* Sa facilité d'utilisation. Cependant, contrairement aux langages C, Go est très facile à utiliser. Son code est logique, et il est facile d’imaginer ce qu’un bloc de code fera avant de l’exécuter.

Ajoutez à cela l'aspect sécurité et vous avez une langue qui semble être là pour rester.

Si vous souhaitez vous familiariser avec Go, que ce soit en tant que langue maternelle ou en tant que langue à ajouter à la boîte à outils de votre développeur, vous avez le choix entre plusieurs cours en ligne. Recherchez un prestataire de cours fiable, qui a une bonne réputation et qui offre un bon support d'apprentissage.

***TypeScript*/*JavaScript***

JavaScript grandit rapidement pour devenir un chef de file du développement Web front-end et back-end. Il peut être utilisé pour tout, de la création de sites Web à la création d'applications mobiles natives, ce qui en fait l'un des langages les plus polyvalents du marché. Cependant, JavaScript présente un inconvénient majeur: il s’agit d’un langage dynamique, ce qui signifie que le dépannage peut être difficile si vous faites des erreurs dans votre code.

TypeScript est une nouvelle version de JavaScript qui autorise le code typé de manière statique en utilisant la même syntaxe et les mêmes conventions que le JavaScript traditionnel. Il est facile à utiliser - surtout si vous connaissez déjà JavaScript - et offre un potentiel énorme en termes de programmation future.

Bien qu’il ne s’agisse pas d’un langage technique, mais plutôt d’une extension de JavaScript, TypeScript est devenu l’un des meilleurs langages de programmation en raison de son énorme potentiel. Rendez-vous sur [**le site Web TypeScript ici**](https://www.typescriptlang.org/) pour plus d'informations.

Si vous souhaitez apprendre le typeScript, vous devriez probablement commencer avec JavaScript traditionnel. [**Le didacticiel JavaScript interactif**](https://fr.bitdegree.org/cours/tutoriel-javascript) est une excellente introduction pour les débutants et couvre tout ce que vous devez savoir.

***Python***

Ahh, Python. Un des langages de programmation les plus anciens et les plus populaires de tous les temps, Python a été classé parmi les meilleurs langages de programmation en raison de sa polyvalence et de ses multiples cas d'utilisation. [**Des sondages récents**](https://insights.stackoverflow.com/survey/2018/#most-loved-dreaded-and-wanted) l'ont classé comme le langage de programmation le plus recherché et le plus apprécié, ce qui suggère qu'il mérite vraiment sa place sur notre liste.

Python est très populaire dans le domaine des sciences, en particulier dans les domaines de l’analyse de données. Il est relativement simple d'écrire un nouveau programme scientifique en Python. Cela en fait une langue d'apprentissage indispensable pour des personnes telles que les chercheurs et les assistants de laboratoire.

Cependant, il est également utilisé pour développer des applications telles que les applications Web et les API, et il est très compatible avec d'autres langages. Python est sans doute le meilleur langage de programmation à apprendre, surtout si vous ne travaillez pas dans un domaine de programmation spécifique. Vous apprendrez très facilement si vous connaissez déjà une autre langue, mais celle-ci reste une excellente langue pour les débutants. En fait, le python est la langue la plus populaire enseignée dans les écoles.

Si vous pensez que Python pourrait vous être utile, consultez [**le cours Apprendre les bases de Python**](https://fr.bitdegree.org/cours/python-en-ligne) sur BitDegree.

**Kotlin**

Kotlin s'est récemment classé comme la deuxième langue la plus aimée au monde, derrière le favori du culte Rust. Il a également reçu une assistance officielle continue de la part d’Android Studio et de Google, ce qui lui confère instantanément l’un des meilleurs langages de programmation à apprendre.

Kotlin a été développé comme une alternative à Java. Il résout un certain nombre de problèmes inhérents au langage Java et commence à devenir le langage de choix pour de nombreuses grandes équipes Android. Parmi les fonctionnalités les plus attrayantes de la langue Kotlin, citons:

* *C'est concis, ce qui signifie que vous devez écrire moins de code pour obtenir le même résultat. Il y a moins de place pour les erreurs et vos programmes seront plus compacts.*
* Il est sûr d’éliminer des classes d’erreur entières - par exemple, les exceptions de pointeur nul n’existent pas dans Kotlin.
* Il est convivial, permettant l’utilisation à la fois des bibliothèques existantes de JVM, Android et de navigateur, ainsi que des outils Java IDE.

Comme vous pouvez le constater, Kotlin est sur le point de reprendre une part importante du marché de Java. Si vous êtes ou souhaitez devenir un développeur Android, il s'agit certainement de l'un des principaux langages de programmation à apprendre, avec une forte demande de développeurs à l'heure actuelle.

***Rust***

Rust est un langage de programmation système dont les cas d'utilisation sont assez étroits. Bien que ce ne soit pas essentiel pour la plupart des programmeurs de l’apprendre, c’est utile pour tous ceux qui envisagent de coder des systèmes de quelque manière que ce soit. Parrainé par Mozilla, Rust semble avoir beaucoup de potentiel de croissance.

[**Une étude récente de Stack**](https://insights.stackoverflow.com/survey/2018/#most-loved-dreaded-and-wanted) a classé Rust au sommet de la liste des langues les plus appréciées pour la troisième année consécutive. Il a également figuré parmi les 10 langues les plus demandées, renforçant ainsi sa position sur notre liste.

Bien que le langage ne soit pas encore très répandu, Rust possède de nombreuses fonctionnalités qui le rendent très favorable. Ceux-ci inclus:

* *Sa vitesse. La rouille est très rapide, ce qui la rend idéale pour le développement Web.*
* C'est sûr. En tant que langage similaire à C et C++, Rust a la particularité d'être nettement plus sûr.
* *C'est sophistiqué. Comparé à d'autres langages modernes comme Go, Rust se distingue par sa facilité d'écriture de code complexe.*

En raison de sa complexité, Rust n’est probablement pas le meilleur langage pour les débutants sans expérience de la programmation. Toutefois, si vous souhaitez approfondir vos connaissances avec un langage idéal pour la programmation système, la rouille est définitivement un bon choix!

***PHP***

PHP, le langage d'Internet. PHP a fait la liste des meilleurs langages de programmation pour un seul fait: il est partout! Plus de 80% des sites Web utilisent au moins une partie du code PHP dans leur conception front-end ou back-end, ce qui signifie que les développeurs PHP auront du travail à faire pour longtemps.

PHP est unique en raison de sa capacité à être intégré au code HTML. Cela signifie qu'au lieu de devoir générer du code HTML comme vous le feriez avec Perl ou C, vous pouvez passer directement du mode PHP à l’extérieur avec des commandes simples.

L'un des principaux avantages de PHP est sa simplicité pour les programmeurs débutants. Il est très facile à apprendre mais offre une large gamme de fonctionnalités pour les développeurs plus avancés. Bien qu'il soit principalement utilisé pour la programmation Web, vous pouvez faire autre chose avec PHP.

Si cela vous intéresse - et cela devrait être le cas -, jetez un œil au cours **[Learn Online de BitDegree](https://www.bitdegree.org/course/php-course)** ou explorez [**le Guide vidéo de devenir un spécialiste de PHP.**](https://fr.bitdegree.org/cours/langage-php)

***Elixir***

L'élixir est probablement la langue la moins connue de notre liste, mais elle a été incluse en raison de son énorme potentiel pour prendre d'assaut le monde de la programmation dans le futur. En tant que langage fonctionnel et dynamique, Elixir a été conçu pour créer des applications totalement évolutives et faciles à gérer. Il est utilisé par le fournisseur de chat populaire Discord et s’est avéré performant sous de très lourdes charges d’utilisateur.

Certaines des fonctionnalités les plus populaires du langage Elixir incluent:

* *Son évolutivité. Comme le prouve son utilisation dans l'application Discord, Elixir est extrêmement facile à faire évoluer et peut gérer un grand nombre d'utilisateurs simultanés.*
* *Sa tolérance aux fautes. Si vous construisez un nouveau programme, les choses iront mal. En tant qu’êtres humains, nous ne sommes pas parfaits et le code parfait est presque impossible à réaliser. Cependant, Elixir est conçu pour gérer les erreurs d'un programme, plutôt que de tomber en panne ou de ne pas s'exécuter.*
* Sa programmation fonctionnelle , qui permet aux développeurs d’écrire du code rapidement et facilement.

Si vous êtes débutant dans l’apprentissage de la programmation, Elixir n’est probablement pas un choix judicieux en raison du manque de ressources et de didacticiels. Cependant, les programmeurs expérimentés pourraient bénéficier de l’ajouter à leur CV car la demande de développeurs d’Elixir devrait augmenter au cours des prochaines années.

### Les contraintes personnelles liées au métier :

* La gestion du stress liée aux délais etc.
* La position assise prolongée
* Mes problèmes musculosquelettiques (cervical, lombaires etc)
* Autres à développer

### Les mots clés ce qui me motive à devenir concepteur ou développeur :

Le présent et le futur de l’humanité est incontestablement déterminé par développement de technologies. La communication avec la « machine » m’intéresse et me passionne depuis toujours ; dans une époque hyper connectée où les smart phones, les tablettes et les ordinateurs font partie indispensable du quotidien de l’homme, j’aimerais investir mes capacités et compétences pour devenir développeur/concepteur potentiellement recherché par les start-ups et entreprises d’innovation, recherche et développement des applications mobiles (notamment le développement iOS).

### Mon expérience en lien avec l’informatique

J’ai créé mon compte GitHub il y a 4 ans où je partage mon code Swift et mes exercices tirés de Playgrounds. J’utilise également les logiciels bureautiques tels que Pages, Numbers, Keynote sur MacOS en parallèle avec Word, Excel, Power Point etc sur WindowsOS. Je réalise des créations graphiques sur Procreate, iPad.

### Les opportunités d’emploi dans ma région

En 2019 j’ai fait environ 200 candidatures spontanées en alternance visant les entreprises les plus renommées en région Rhône Alpes dans un cadre de formation OpenClassrooms que n’a pas été abouti.

Pour la suite je viserais les grandes entreprises et les start-ups de multimédia, marketing, publicité, récréation, culture ainsi que les établissements d’état en région Rhône Alpes et, pourquoi pas, en Suisse, au RU, aux EU, etc

**Compétences mises en oeuvre :**

* Découvrir le métier
* Créer un document HTML
* Mettre en forme un document HTML avec CSS

**Réalisation d'une fiche métier**

À partir du cahier des charges rédigé par le chef de projet en fonction des besoins du client, le développeur / la développeuse analyse les besoins, choisit la solution technique la mieux adaptée et développe les fonctionnalités de l'application. Pour cette dernière étape, il écrit des lignes de code.

Le développeur peut aussi apporter des solutions aux problèmes présents dans une application en production. Dans ce cas, le développeur procède au diagnostic et à l'implémentation des correctifs, sans interruption du fonctionnement de l'application.

A partir de cette description générale du métier de développeur et à partir de recherches sur le web que vous effectuerez, vous devez créer une fiche métier exposant **votre vision** du métier de développeur. Vous réaliserez cette fiche métier au format Word (ou équivalent) puis, dans un 2ème temps, au format HTML.

**Travail à réaliser**

Créez un répertoire pour votre projet.

Dans ce répertoire,

1. Rédigez et complétez votre fiche métier dans un document **Word** ou équivalent.
2. Validez votre fiche métier avec vos formateurs.
3. Créez un fichier metier.html qui contiendra votre fiche métier au format HTML.
   1. Créez la structure HTML
   2. Créez l'entête de page et mettez la en forme avec CSS
   3. Ajoutez la 1ère section et mettez la en forme avec CSS
   4. Complétez section par section

Lorsque vous avez terminé, validez votre code HTML et CSS avec vos formateurs.

Note: Tous les fichiers de votre site (html, css, images) doivent être situés dans le répertoire de votre projet. Vous pouvez ajouter autant de fichiers et sous-répertoires que nécessaire.

**A vous de jouer**

Vous avez la liberté de mettre en forme votre fiche comme vous le souhaitez tant que la structure demandée est rigoureusement respectée :)