

**Mémoire de projet de fin d’année**

Option

**Filière Data Engineer**

Sujet

Analysis on data covid19 using web scraping (Python)

Soutenu par : Sous la direction de :

**BENJARNIJ Hajar Mme AYACHE Meryem**

**BENIDDER Latifa Mme EL ASRI Ikram**

Année universitaire : 2021/2022

Remerciements

Le travail présenté dans ce rapport a été effectué dans le cadre du projet Analysis on Covid-19 Data Web Scraping à l'institut national des postes et télécommunications (INPT). En terme de ce projet, nous tenons à exprimer notre profonde gratitude et notre immense respect à Mme Ikram El Asri et Mme Ayach Meryem , notre chère professeures pour leurs disponibilité, leurs avis éclairés, et leurs judicieux conseils. Nos vifs remerciements accompagnés de toute notre gratitude s ' adressent également à Mr. Dahchour, le directeur adjoint des études, qui assure le bon déroulement ainsi que la bonne organisation des cours . Nous exprimons aussi notre gratitude à Mr. Baina, notre chef de filière (DATA Engineering) qui nous a accompagnés durant toute cette année. Avec beaucoup d'égard, nous ne manquerons pas d'exprimer notre grande reconnaissance à tous les enseignants et administrateurs de notre cher institut.

Table des matières

[Introduction générale 6](#_Toc105934752)

[Chapitre 1 : Covid19 pandémie 8](#_Toc105934753)

[I- Covid19 pandémie 9](#_Toc105934754)

[1- Covid19 9](#_Toc105934755)

[**2-** La protection contre le coronavirus 9](#_Toc105934756)

[**3-** Principaux modes de transmission 9](#_Toc105934757)

[**4-** Le début de la pandémie 9](#_Toc105934758)

[**5-** Résultats : 10](#_Toc105934759)

[II- L’impact de la pandémie sur le monde 10](#_Toc105934760)

[1- Secteur économique 10](#_Toc105934761)

[**Impact de la pandémie sur l’économie marocaine** 10](#_Toc105934762)

[2- Secteur social 11](#_Toc105934763)

[III- Vaccination 14](#_Toc105934764)

[1- Types de vaccin 14](#_Toc105934765)

[2- Statistiques sur la vaccination 14](#_Toc105934766)

[3- Efficacité et effet sur la propagation de covid19 15](#_Toc105934767)

[4- Conclusion 16](#_Toc105934768)

[Chapitre 2 : Web Scraping 17](#_Toc105934769)

[1- Définition dU web scraping 18](#_Toc105934770)

[2- Les types de web scraping 18](#_Toc105934771)

[3- L'utilité de web scraping 19](#_Toc105934772)

[4- Le web scraping est-il légal 19](#_Toc105934773)

[**Chapitre 3 : Résultats et analyses** **I- Outils techniques** 21](#_Toc105934774)

[**I- Outils techniques** 22](#_Toc105934775)

[1- Langages de programmation 22](#_Toc105934776)

[Web scraping avec Python 22](#_Toc105934777)

[Web scraping avec Node.js 22](#_Toc105934778)

[Web scraping avec PHP 22](#_Toc105934779)

[Web scraping avec Ruby 23](#_Toc105934780)

[Scraper en C ou C++ 23](#_Toc105934781)

[scraper avec R 25](#_Toc105934782)

[2- Les libraires python pour le web scraping 25](#_Toc105934783)

[Scrapy 25](#_Toc105934784)

[Pyspider 26](#_Toc105934785)

[Requests 26](#_Toc105934786)

[BeautifulSoup 27](#_Toc105934787)

[3- Les algorithmes utilisés 28](#_Toc105934788)

[II- Résultats 33](#_Toc105934790)

[1- Extraction des données 33](#_Toc105934791)

[2- Visualisation 35](#_Toc105934792)

[III-Analyse 38](#_Toc105934793)

[1- Les pays les plus touchés 38](#_Toc105934794)

[2- Relation entre population et cas contaminées – Raisons politiques et économiques 39](#_Toc105934795)

[Conclusion 40](#_Toc105934796)

# Introduction générale

La pandémie du covid19 a constituée, et continuer d’être, une menace existentielle avec un impact négatif inédit dans l’Histoire récente de l’Humanité. Cette crise sanitaire a touchée divers aspects de la vie communautaire, et s’est rapidement transformer en une crise économique et sociale dont les répercussions vont se prolonger même après avoir surmonté la pandémie.

En 2019 la maladie a apparue la première fois en Chine, après, due aux voyages et contacts, le virus a propagé dans tout le monde pour dépasser un nombre des cas contaminées 7.030.828. cette crise a été la raison de décès de plusieurs personnes qui a dépassé 6 million.

La COVID-19 est partout, littéralement, et en 2020, sa propagation et son impact sur les populations de toutes les régions du monde ont conduit à une crise mondiale d'une portée et d'une proportion sans précédent. ONU Info revient sur certains des moments clés des 12 derniers mois.

Cette année, l'éducation des enfants du monde entier a été gravement perturbée, alors que les écoles ont lutté pour faire face aux fermetures et réouvertures répétées et au passage à la scolarité en ligne, lorsque celle-ci était possible. Les enfants défavorisés ont cependant été les plus touchés par les mesures d'urgence.

En 2020, le COVID-19 a atteint presque tous les pays et touché plus de 50 millions de personnes dans le monde L’administration publique a été amenée à travailler dans un contexte d'incertitude totale et à opérer des arbitrages difficiles face aux défis que représente le virus sur le plan sanitaire, économique et social. Au printemps 2020, plus de la moitié de la population mondiale a subi un confinement assorti de mesures fortes visant à enrayer la propagation du virus. Au-delà de la tragédie humaine et sanitaire que représente le coronavirus, il est maintenant largement admis que la pandémie a provoqué la crise économique la plus grave depuis la Seconde Guerre mondiale.

Toutefois, Le degré de contamination et d’impact de covid19 sur les pays du monde n’étaient pas les mêmes. L’objectif de ce projet est, d’abord, la visualisation de l’effet de la pandémie dans le monde à savoir les pays les plus touchés, le nombre des décès et de guérissons maximal et minimal. Ensuite, Analyser ses observation et essayer de trouver ses lien avec l’état économique et politique de chaque pays ainsi la localisation géographique.

# 

# Chapitre 1 : Covid19 pandémie

# Covid19 pandémie

## Covid19

La plupart des personnes infectées par le virus présentent une maladie respiratoire d’intensité légère à modérée et se rétablissent sans avoir besoin d’un traitement particulier. Certaines, cependant, tombent gravement malades et ont besoin de soins médicaux. Les personnes âgées et celles qui ont un problème médical sous-jacent, tel qu’une maladie cardiovasculaire, un diabète, une maladie respiratoire chronique ou un cancer, ont plus de risques de présenter une forme grave. N’importe qui, à n’importe quel âge, peut contracter la COVID-19 et tomber gravement malade ou en mourir.

## La protection contre le coronavirus

La meilleure façon d’éviter et de ralentir la transmission est d’être bien informée sur la maladie et sur la manière dont le virus se propage. Protégez-vous et protégez les autres contre l’infection en maintenant une distance d’au moins un mètre avec les autres, en portant un masque correctement ajusté et en vous lavant les mains fréquemment à l’eau et au savon ou avec une solution hydro alcoolique. Faites-vous vacciner quand vient votre tour et suivez les recommandations locales.

## Principaux modes de transmission

Le virus peut se propager par l’intermédiaire des gouttelettes de salive ou de sécrétions nasales émises par une personne infectée quand elle tousse, éternue, parle, chante ou respire. Il est donc important d’appliquer les règles d’hygiène respiratoire, par exemple en se couvrant la bouche et le nez avec le pli du coude lorsque l’on tousse, et si l’on ne se sent pas bien, de rester chez soi et de s’isoler jusqu’à ce qu’on soit rétabli.

## Le début de la pandémie

L'Organisation mondiale de la santé (OMS) alerte dans un premier temps la République populaire de Chine et ses autres États membres, puis prononce l'état d'urgence de santé publique de portée internationale le 30 janvier 2020.

Le 11 mars 2020, l'épidémie de Covid-19 est déclarée pandémie5,6 par l'OMS qui demande des mesures de protection essentielles pour prévenir la saturation des services de soins intensifs7 et pour renforcer l'hygiène préventive (suppression des contacts physiques, bises et poignées de mains, fin des attroupements ainsi que des déplacements et voyages non indispensables, promotion du lavage des mains, mise en application de quarantaine, etc.). Pour freiner la formation de nouveaux foyers de contagion et préserver les capacités d'accueil de leurs hôpitaux, de nombreux pays décident des mesures de confinement, la fermeture de leurs frontières et l'annulation des manifestations sportives et culturelles. Ces décisions ont des conséquences économiques, sociales et environnementales et font peser des incertitudes et des craintes sur l'économie mondiale et sur l'éducation, la santé et les droits fondamentaux des populations.

## Résultats :

Depuis le début de corona 532 231 616 cas infectés ont été enregistrés dans le monde entier, et de ce nombre il t en 503 194 840 qui se sont rétablis de corona et 6 312 181 qui sont morts à cause de cette épidémie.

# II- L’impact de la pandémie sur le monde

## 1- Secteur économique

**Impact de la pandémie sur l’économie marocaine**

Le Maroc mène des efforts pour, d’un côté, contenir la propagation du virus sur son sol, et, de l’autre, sauver son économie nationale, dont des pans entiers ont dû brusquement baisser le rythme ou céder complètement à l’arrêt. Les conséquences de la pandémie sur l’économie marocaine sont très difficiles à évaluer, car nul ne sait à quelle date elle prendra fin. Mais d’ores et déjà, le Haut-commissariat au plan a dévoilé le 22 avril une étude sur l’impact immédiat de la crise :

* La croissance économique nationale serait amputée de 8,9 points, au deuxième trimestre 2020, par rapport à son évolution d’avant crise Covid-19, au lieu de -3,8 points prévu au 7 avril, Cette baisse représenterait une perte globale potentielle d’environ 29,7 milliards de DH pour la première moitié de 2020, au lieu de 15 milliards de DH prévue au 7 avril
* Près de 142 000 entreprises, soit 57 % du tissu économique, ont arrêté définitivement ou temporairement leurs activités. Les TPE sont les plus touchées : elles représentent 72 % des entreprises en difficulté, tandis que 26 % sont des PME et seulement 2 % des grandes entreprises.
* La croissance de la demande étrangère adressée au Maroc a été révisée à la baisse, pour atteindre -12,5% au deuxième trimestre 2020, au lieu de -6% prévu au 7 avril, suite au fléchissement attendu des importations des principaux partenaires commerciaux du Royaume
* L’aggravation du déficit s’explique principalement par l’augmentation des dépenses sociales et économiques liées à Covid19 et la baisse des recettes fiscales, en particulier de l’impôt sur les sociétés, par conséquent, la dette de l’administration centrale pourrait culminer à 73% du PIB en 2020.
* L’investissement, quant à lui, poursuivrait son repli de -26,5% par rapport au deuxième trimestre 2019,
* Un net ralentissement des exportations, des recettes touristiques et des envois de fonds est prévu, car la pandémie perturbe le commerce et les chaînes de valeur mondiale.
* L’incertitude globale sur le marché du travail conduit à la perte d’emplois et de revenus, sans savoir à quel moment ils pourraient être rétablis
* 27 % des entreprises ont réduit temporairement ou définitivement leurs effectifs, ce qui représente près de 726 000 postes, soit 20% de la main d’œuvre des entreprises organisées, hors secteur financier et agricole.

Certains secteurs ont montré des signes précoces de vulnérabilité tels que le tourisme, les transports et la logistique dans les chaînes d’approvisionnement, mais aussi – et plus difficile à mesurer – le secteur informel transversal. Comme attendu, le secteur le plus touché est celui de l’hébergement et de la restauration, dont 89 % des entreprises qui sont à l’arrêt. Elles sont 76 % dans le secteur des industries textiles et du cuir et 73 % dans celui des industries métalliques et mécaniques et 60 % dans la construction. La pêche, les mines et l’agro-industrie sont les secteurs les moins touchés par la crise sanitaire, comptant respectivement 24 %, 32 % et 34 % d’entreprises à l’arrêt.

## 2- Secteur social

En plus des demeures économiques que la pandémie a laissée, on trouve aussi l’impact de la COVID-19 qui n’a pas été le même pour tous. Les personnes qui souffrent des taux les plus élevés et des pires conséquences de l’infection sont celles qui souffraient de discrimination, de marginalisation et de pauvreté bien avant la pandémie. Cet impact inégal est ancré dans une réalité préexistante caractérisée par une injustice structurelle et des iniquités colossales. Pour ceux qui ont choisi de fermer les yeux, la pandémie les oblige à les ouvrir grands. Comme certains l’ont fait remarquer, la pandémie est une radiographie qui rend visible ce qui est habituellement caché. C’est une loupe qui met en relief nos défaillances sociales.

##### Pauvreté et inégalités

* Selon la [Banque mondiale](https://fr.wikipedia.org/wiki/Banque_mondiale), 150 millions de personnes dans le monde ont plongé dans la pauvreté à cause des effets de la crise sanitaire.
* D'après le magazine économique [*Forbes*](https://fr.wikipedia.org/wiki/Forbes_(magazine)), entre le 18 mars et le 19 mai 2020 « la valeur nette des 600 plus grosses fortunes américaines a augmenté de 434 milliards de dollars », une augmentation de 15 %. Début décembre, cette augmentation s'élève à plus de 1 000 milliards de dollars.
* Selon [Oxfam](https://fr.wikipedia.org/wiki/Oxfam_France), il a fallu neuf mois aux plus riches pour retrouver leur niveau de fortune d'avant la pandémie, il faudra plus de dix ans aux personnes les plus pauvres pour s'en relever. Selon la [Banque mondiale](https://fr.wikipedia.org/wiki/Banque_mondiale), si les pays du monde « interviennent sans attendre pour réduire les inégalités », la pauvreté pourrait revenir à son niveau pré-Covid « en seulement trois ans »
* Selon [La Presse](https://fr.wikipedia.org/wiki/La_Presse_(Montr%C3%A9al)), les leaders Afro-Américains demandent une révision des politiques systémiques qui ont rendu leur communauté plus vulnérable à la Covid-19

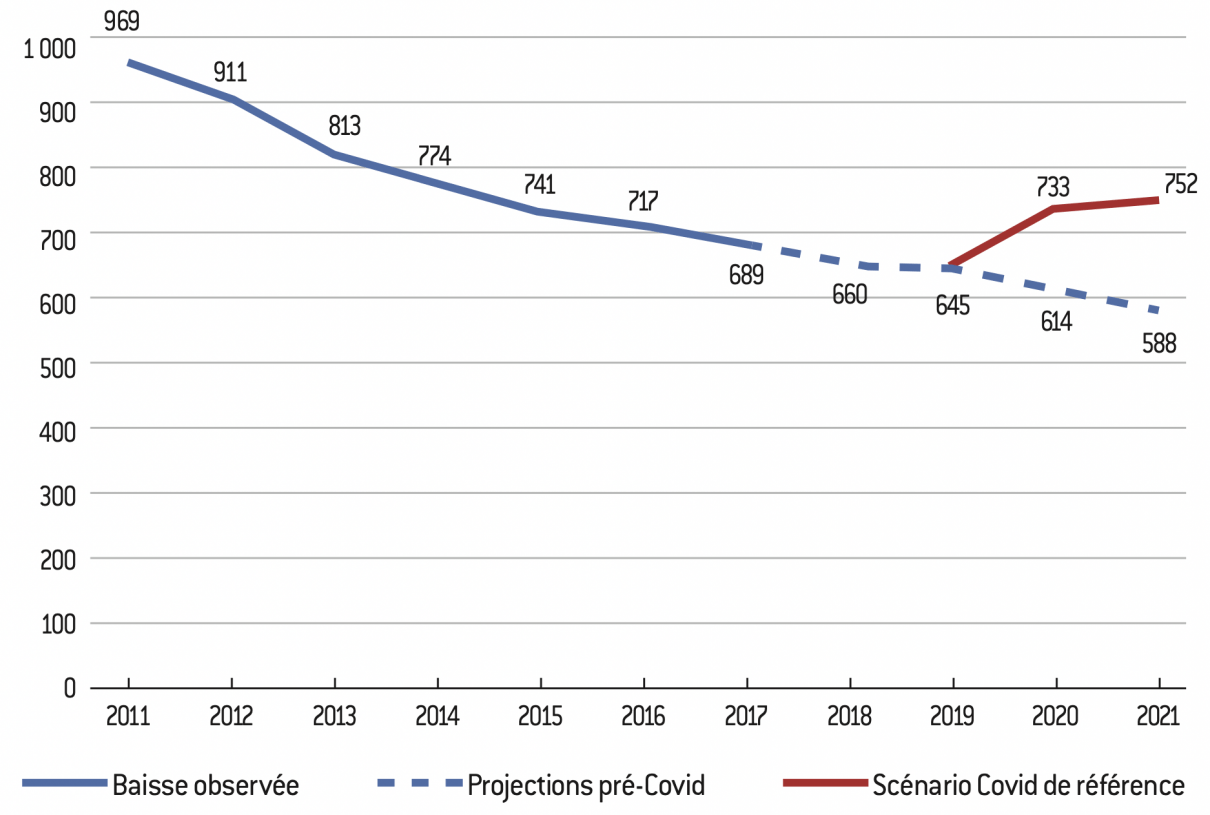


Figure 1 : Projections d’évolutions de la pauvreté globale, 2011-2021 (en millions d’individus)

##### Effets sur la vie privée et les modes de sociabilité

Les mesures de confinement ont profondément transformées les modes de sociabilité des habitants. Les liens sociaux subjectifs de la population ont été relativement peu compromis par les mesures de distanciation physique mises en place pour lutter contre la pandémie.

##### Violences domestiques

À fin mars, le nombre de signalements de [violences conjugales](https://fr.wikipedia.org/wiki/Violence_conjugale) est en hausse de 30 %.

##### Conséquences scolaires

Les mesures de fermeture des écoles prises dans de nombreux pays ont des conséquences particulièrement graves pour les enfants des milieux défavorisés. Le confinement et l'école a brisé la relation entre les enseignants et les élèves et fait ressentir une fracture numérique des élèves d’écoles, de collèges, de lycées et d’universités n’ayant pas d’ordinateur ou pas assez performant ou ayant une trop mauvaise connexion ou connexion limitée avec petite enveloppe de données.

# III- Vaccination

## Types de vaccin

En décembre 2020, plus de 200 vaccins candidats contre la COVID-19 étaient en cours de développement. Sur ceux-ci, au moins 52 vaccins candidats sont au stade des essais sur l’homme. Quelques types de vaccin ont été utilisés dans le monde :

* **Sinopharm** : Vaccin entier inactivé anti-covid 19 cultivé sur cellules Vero. Autres noms : BIBP, BBIBP inactivated COVID-19 Vaccine, SINOPHARM BIBP COVID-19 vaccine, HB02 COVID-19 Vaccine.
* **Sinovac**: Vaccin anti-covid 19 entier inactivé avec adjuvant. Autre nom : Sinovac COVID-19 Vaccine.
* **Vaccin Spikevax®** : Le vaccin est approuvé pour les personnes de 6 ans et plus. Son innocuité et son efficacité chez les jeunes de moins de 6 ans n'ont pas encore été démontrées.
* **Janssen**: Le vaccin est approuvé pour les personnes de 18 ans et plus. Son innocuité et son efficacité chez les jeunes de moins de 18 ans n'ont pas encore été démontrées.

## Statistiques sur la vaccination

Dans le monde entier 60% a pris la vaccination contre covid19.voilà quelques statistiques sur le pourcentage des gens vaccinés dans certains pays.

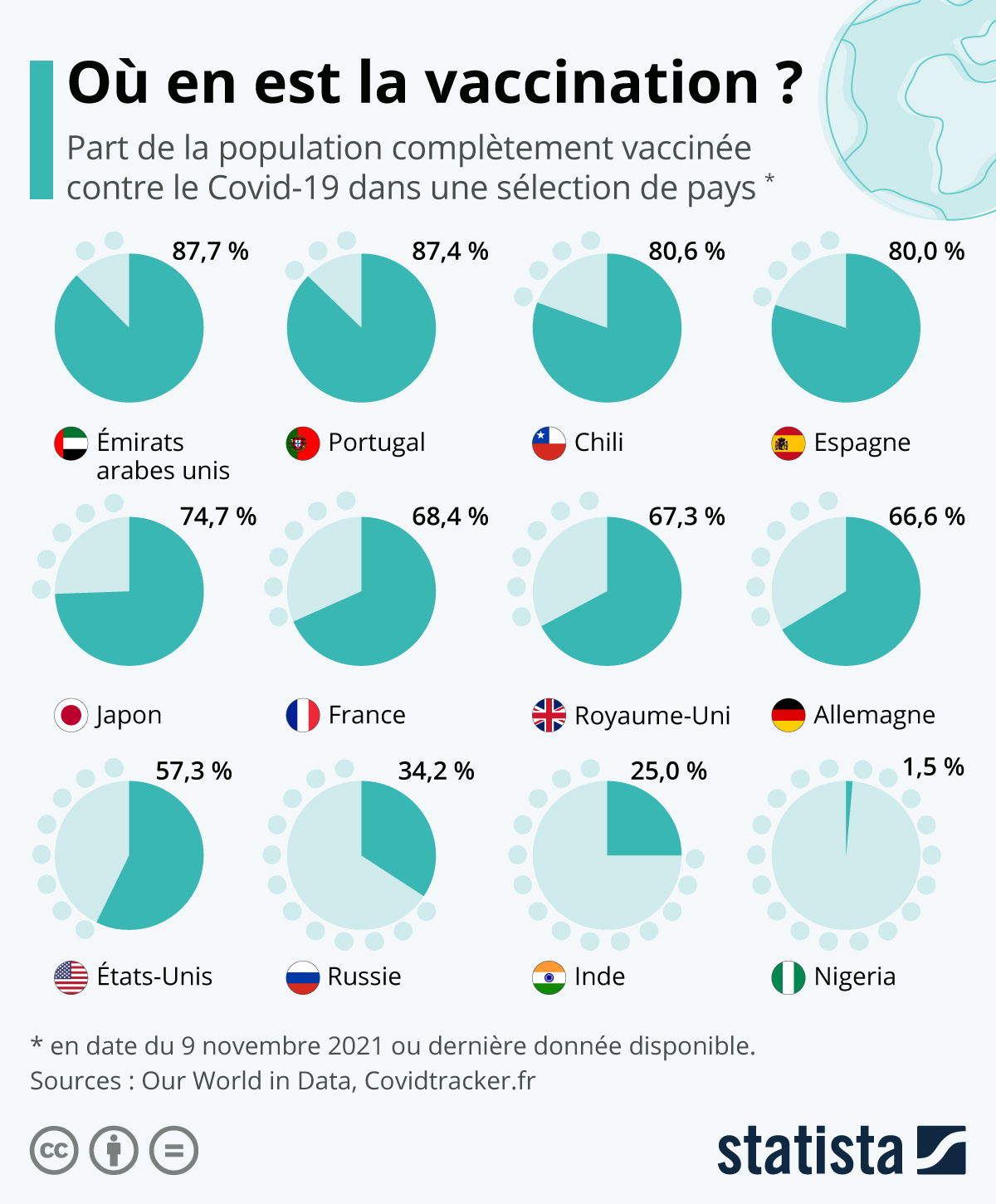


Figure 2 : Part de la population totalement vaccinés dans certains pays

## Efficacité et effet sur la propagation de covid19

En effet, la vaccination protège l'individu afin qu'il ne développe pas une forme grave de la maladie grâce aux [anticorps](https://www.futura-sciences.com/sante/dossiers/medecine-vaccins-leurs-effets-notre-systeme-immunitaire-696/page/7/) présents, boostés par le [vaccin](https://www.futura-sciences.com/sante/definitions/medecine-vaccin-4060/). Les [données d’efficacité](https://vaccination-info-service.fr/Les-maladies-et-leurs-vaccins/COVID-19#panel-group-1) du vaccin Pfizer-BioNTech montrent qu'aujourd'hui, en France, le risque d'hospitalisation lié à la Covid-19 est réduit de 87 % chez les personnes de plus de 75 ans vaccinées. Parallèlement, une réduction de 91 % du risque de décès est observée dans la même population.

De fait, le [virus](https://www.futura-sciences.com/sante/dossiers/medecine-savoir-plus-virus-236/) sera présent en moindre quantité dans l'organisme mais pourra potentiellement être transmis d'une personne à l'autre. Sur d'autres modèles de maladies à transmission virale (comme le [VIH](https://www.futura-sciences.com/sante/dossiers/medecine-sida-vaincre-vih-1696/)), les scientifiques admettent aujourd'hui que la baisse de la [charge virale](https://www.futura-sciences.com/sante/definitions/medecine-charge-virale-3129/) diminue significativement la [contagiosité](https://www.futura-sciences.com/sante/questions-reponses/virus-sont-virus-plus-contagieux-13140/). On peut donc raisonnablement estimer qu'avoir moins de [virus](https://www.futura-sciences.com/sante/definitions/medecine-virus-291/), c'est être moins infectieux.

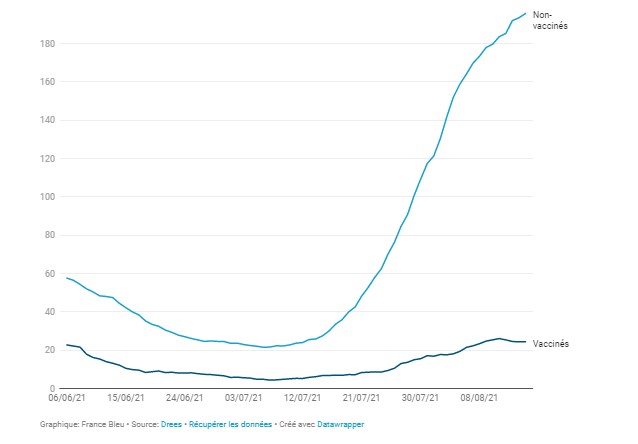


Figure 3 : la variation des cas contaminés dans le cas vacciné et le cas non vacciné.

## Conclusion

Depuis l’apparition de covid19, le mode de vie a changé. La pandémie a laissée des effets sociaux et économiques dans le monde entier. Plus de 6 Millions de décès à cause de celui la. La vaccination a contribué plus ou moins à faire face à ce virus, mais sont impact sur les différents aspects de la vie reste important. Vue que la distribution différente des cas dans les différents pays du monde, ce projet a pour objectif de visualiser et analyser cette distribution en fonction des facteurs tels que la population et la localisation géographique.

# Chapitre 2 : Web Scraping

## Définition dU web scraping

Le Web scraping (de l’anglais scraping = « gratter/racler ») consiste à extraire des données de sites Internet et à les enregistrer afin de les analyser ou de les utiliser de toute autre façon. Le scraping permet de collecter des informations de nature bien différente. Il peut par ex. s’agir de coordonnées comme des adresses e-mail ou des numéros de téléphone, mais aussi de mots-clés individuels ou d’URL. Ces informations sont alors rassemblées dans des bases de données locales ou des tableaux.

## Les types de web scraping

Le scraping comprend différentes fonctionnalités, mais on opère généralement une distinction entre le scraping manuel et automatique. Le scraping manuel désigne le fait de copier et insérer manuellement des informations et des données. On peut le comparer avec le fait de découper et rassembler des articles de journaux. Le scraping manuel est uniquement effectué lorsque l’on souhaite trouver et enregistrer des informations de façon sporadique. Il s’agit d’un processus très laborieux qui est rarement appliqué pour de grandes quantités de données. Dans le cas du scraping automatique, on utilise un logiciel ou un algorithme qui explore plusieurs sites Internet afin d’extraire des informations. Un logiciel spécifique est utilisé en fonction de la nature du site Internet et du contenu. Dans le scraping automatique, on distingue différentes méthodes :

• Les analyseurs syntaxiques : un analyseur syntaxique est utilisé pour convertir le texte en une nouvelle structure. Dans le cas de l’analyse d'un HTML par exemple, le logiciel lit le document HTML et enregistre les informations. L’analyse d’un DOM utilise l’affichage des contenus dans le navigateur côté client pour extraire les données.

• Les robots : un robot est un logiciel réalisant des tâches spécifiques et les automatisant. Dans le Web harvesting, les robots sont utilisés pour explorer automatiquement des sites Internet et collecter des données.

• Le texte : les personnes sachant utiliser la Command Line peuvent utiliser les instructions Unix grep pour explorer le Web à la recherche de certains termes dans Python ou Perl. Il s’agit d’une méthode très simple pour obtenir des données qui requiert toutefois davantage de travail que lorsqu’on utilise un logiciel.

## L'utilité de web scraping

Le Web scraping est utilisé pour une multitude de tâches. Il permet par ex. de récolter rapidement des données de contact ou des informations spécifiques. Dans le domaine professionnel, le Web scraping est souvent utilisé pour accéder à des avantages par rapport à des concurrents. Le harvesting de données permet à une société de consulter tous les produits d’un concurrent et de les comparer avec ses propres produits. Le Web scraping apporte également une valeur ajoutée pour les données financières :

Les informations sont lues depuis un site Internet externe, transférées dans un tableau et peuvent ensuite être analysées et traitées.

Google constitue un bon exemple de Web scraping. Le moteur de recherche utilise cette technologie afin d’afficher des informations météorologiques ou des comparatifs de prix pour des hôtels et des vols. Parmi les comparateurs de prix, nombreux sont ceux à également utiliser le Web scraping afin d’afficher des informations de nombreux sites Internet et prestataires.

## Le web scraping est-il légal

Le scraping n’est pas toujours légal et les « scrapers » doivent tout d’abord respecter les droits d’auteur des sites Internet. Pour certaines boutiques en ligne et prestataires, le Web scraping a des conséquences très négatives, par ex. lorsque le classement d’une page est affecté par des aggravateurs. Aussi n’est-il pas rare de voir une entreprise poursuivre en justice un comparateur afin de mettre un terme au Web scraping. Une affaire de ce type a été présentée à la Cour d’appel de Francfort en 2009, qui a toutefois statué qu’une compagnie aérienne devait permettre aux comparateurs de réaliser du Web scraping car leurs informations étaient librement accessibles. La compagnie aérienne a néanmoins la possibilité d’installer des mesures techniques pour empêcher le scraping. Le scraping est donc légal lorsque les données extraites sont librement accessibles par des tiers sur le Web. Pour vous assurer d’être dans votre bon droit en faisant du Web scraping, respectez les règles suivantes:

• Consultez et respectez les droits d’auteur. Lorsque les données sont protégées par des droits d’auteur, elles ne peuvent pas être publiées à un autre endroit. • Les exploitants de site Internet sont en droit d’installer des processus techniques empêchant le Web scraping. En aucun cas, ces processus ne doivent être contournés

• Lorsque l’utilisation des données est liée à une connexion utilisateur ou à un contrat d’utilisation, ces données ne doivent pas faire l’objet d’un Web scraping.

• Le fait de masquer de la publicité, des conditions générales d’utilisation ou des clauses de non responsabilité avec une technologie de scraping est interdit. Même si le scraping est autorisé dans de nombreux cas, il peut être utilisé de façon abusive à des fins destructrices, voire illégales. Cette technologie est par exemple souvent utilisée pour envoyer des spams. Grâce à elle, les « spammers » peuvent par ex. collecter des adresses e-mail et envoyer des messages spam à ces destinataires.

## Chapitre 3 : Résultats et analyses I- Outils techniques

## 1- Langages de programmation

## Web scraping avec Python

Le framework BeautifulSoup et la librairie Scrapy sont les deux outils de référence pour le web scraping en Python depuis respectivement 2004 et 2008. Alors que ces solutions permettent de mettre en place facilement un programme de scraping puissant, elles peuvent également s’interfacer avec d’autres modules pour gérer spécifiquement certaines étapes : crawling, fetching, parsing, traitement, visualisation… Python est généralement considéré comme étant un langage facile à apprendre car sa syntaxe est simple et lisible. Depuis de nombreuses années et peut-être encore aujourd’hui, Python est sans doute le langage le plus recommandé pour le web scraping. Cette position est cependant en train de changer avec l’évolution de Node.js et l’arrivée de nouveaux outils.

## Web scraping avec Node.js

Depuis l’arrivée de Puppeteer en 2017, Node s’impose également progressivement comme le meilleur langage pour le web scraping. Puppeteer permet de contrôler très facilement un navigateur Chrome/Chromium programmatiquement en mode “headless”, ce qui permet d’interagir avec tout le JavaScript, de paralléliser les requêtes sur différents onglets etc. Une alternative, Playwright existe et est cross-browser. Node propose une rapidité inégalable, même face à Python et est très simple à utiliser. Des solutions cloud comme AWS Lambda permettent de déployer des scripts Node à grande échelle en quelques clics.

## Web scraping avec PHP

On ne présente plus PHP qui est le langage serveur utilisé par près de 8 sites sur 10 dans le monde selon W3Tech. PHP propose des outils très efficaces pour le scraping telles que les librairies Guzzle ou Goutte, une solution complète portée par le créateur de Symfony. PHP n’est généralement pas recommandé pour le web scraping : il est lent, ne peut pas exécuter JavaScript (donc on oublie les sites contenant beaucoup d’animations JavaScript pour afficher ou masquer des éléments du DOM), ne gère pas la parallélisation des requêtes, le multithreading, les comportements asynchrones… c’est, en tout cas, ce que vous diront ses détracteurs. Si je vous disais que plus de 80% de mes programmes de scraping qui tournent quotidiennement sont codés en PHP, vous changeriez d’avis ? PHP est selon moi l’un des langages les plus faciles pour le web scraping et aussi des plus simples à déployer à petite échelle. À l’inverse d’autres technologies, PHP n’utilise pas de navigateur headless et effectue plutôt ses requêtes via cURL. Le code JavaScript des sites ciblés n’est donc pas exécuté par le script de web scraping. Cela lui permet de ne pas être détecté par les solutions de protection anti scraping qui se basent sur JavaScript. De plus, vous pouvez très facilement imiter les requêtes AJAX permettant d’afficher du contenu sur des sites dynamiques avec PHP et Postman.

## Web scraping avec Ruby

Ruby est cependant limité par sa lenteur d’exécution, en partie due à son statut de langage compilé (comme C et C++). Le multithreading (parallélisation des requêtes sur plusieurs fils d’exécution du processeur) y est supporté mais peu efficace, la consommation de ressources peut donc s’avérer importante. Enfin, le scraping de sites “full JS” ou avec beaucoup de contenu généré dynamiquement sera plus difficile que sur d’autres langages, et le déploiement à grande échelle très coûteux. C’est pourquoi Ruby n’est pas le langage le plus recommandé pour le web scraping.

## Scraper en C ou C++

Les sempiternels C et C++ sont des langages fiables et performants et disposent de tous les outils nécessaires au web scraping : libcurl pour télécharger (fetching) les pages HTML, libtidy pour valider le XML et libxml pour traiter (parsing) le contenu et extraire les données voulues. Paralléliser les requêtes de votre programme pour le rendre plus rapide et performant est d’ailleurs très facile dans ces langages. En revanche, le déploiement d’un programme de web scraping en C ou C++ nécessitera bien plus d’efforts qu’en Node, PHP ou Python, surtout à grande échelle. De plus, ces langages sont à typage statique. Bien que le typage statique puisse être appréciable lorsque vous développez votre programme, dans le cas du web scraping, vous n’avez pas la main sur ce que va vous retourner la page web. Dans ce cas, le typage dynamique permet d’éviter des erreurs pendant l’exécution du programme. Ainsi, C et C++ ne sont généralement pas des langages recommandés pour le web scraping.

## scraper avec R

R est le langage préféré des statisticiens, permettant de créer des analyses et statistiques complexes sur de grands volumes de données avec une très bonne performance. Il s’agit du langage qui propose le plus grand nombre de modules dédiés à l’analyse statistique mais R est également utilisé pour le machine learning et propose de très belles visualisations. Il y a donc un intérêt certain à coder le web scraping dans le même langage, d’autant plus que R dispose de tous les modules nécessaires tels que rcrawler pour le crawling, rvest pour le parsing. R a la réputation d’être assez difficile à apprendre, surtout, et c’est assez paradoxal, pour les développeurs formés à d’autres langages. R est en effet prévu pour les analystes, statisticiens et mathématiciens qui n’ont pas d’expérience en programmation. Scraper est considéré comme une technique avancée en R alors qu’elle est très abordable en Node ou en Python par exemple. R est donc uniquement recommandé pour le web scraping si vous maîtrisez déjà ce langage.

## 2- Les libraires python pour le web scraping

Python est le langage de programmation le plus populaire pour coder des web scrapers en raison de sa syntaxe simple, de sa courbe d’apprentissage et du nombre de bibliothèques disponibles qui facilitent le travail des développeurs. Nous présenterons dans cette section quelques librairies Python permettant d’effectuer le Webscraping.

## Scrapy

Scrapy est un Framework python open-source conçu spécifiquement pour le scraping web par les cofondateurs de Scrapinghub, Pablo Hoffman et Shane Evans. Vous vous demandez peut-être « Qu’est-ce que cela signifie ?« . Cela signifie que Scrapy est une solution de web scraping à part entière qui vous décharge d’une grande partie du travail de construction et de configuration de vos spiders, et surtout, il gère de manière transparente les cas limites auxquels vous n’avez probablement pas encore pensé. Quelques minutes après l’installation du Framework, vous pouvez disposer d’un spider entièrement opérationnel pour scraper le web. Dès le départ, les spiders Scrapy sont conçus pour télécharger du HTML, analyser et traiter les données et les enregistrer dans des formats de fichiers CSV, JSON ou XML. Il existe également un large éventail d’extensions et d’intergiciels intégrés conçus pour gérer les cookies et les sessions, ainsi que des fonctionnalités HTTP telles que la compression, l’authentification, la mise en cache, les agents utilisateurs, le fichier robots.txt et la restriction de la profondeur d’exploration. L’un des plus grands avantages de l’utilisation du framework Scrapy est qu’il est construit sur Twisted, une bibliothèque de mise en réseau asynchrone. Cela signifie que les Scrapy spiders n’ont pas à attendre pour faire des requêtes une par une. Au lieu de cela, ils peuvent effectuer plusieurs requêtes HTTP en parallèle et analyser les données au fur et à mesure qu’elles sont renvoyées par le serveur. Cela augmente considérablement la vitesse et l’efficacité d’un scraper. Un petit inconvénient de Scrapy est qu’il ne gère pas directement JavaScript comme Selenium. Cependant, l’équipe de Scrapinghub a créé Splash, un navigateur headless facile à intégrer, léger et scriptable, spécialement conçu pour le web scraping. La courbe d’apprentissage de Scrapy est un peu plus raide que celle, par exemple, de l’utilisation de BeautifulSoup. Cependant, le projet Scrapy dispose d’une excellente documentation et d’un écosystème extrêmement actif de développeurs sur GitHub et StackOverflow qui publient en permanence de nouveaux plugins et vous aident à résoudre les problèmes que vous rencontrez. Si vous souhaitez créer votre première araignée Scrapy, n’hésitez pas à consulter le site Web du projet Scrapy.

## Pyspider

PySpider est un autre outil de web scraping que vous pouvez utiliser pour écrire des scripts en Python. Contrairement à Scrapy, il peut produire un rendu du JavaScript et, en tant que tel, ne nécessite pas l’utilisation de Selenium. Cependant, il est moins mature que Scrapy, qui existe depuis 2008 et dispose d’une meilleure documentation et d’une meilleure communauté d’utilisateurs. Cela ne rend pas PySpider inférieur. En fait, PySpider est livré avec des fonctionnalités inégalées telles qu’un éditeur de script en interface Web .

## Requests

Requests est une bibliothèque HTTP qui permet d’envoyer facilement des requêtes HTTP. Elle est construite sur la base de la bibliothèque urllib. C’est un outil robuste qui peut vous aider à créer des racleurs web plus fiables. Il est facile à utiliser et nécessite moins de lignes de code. Il est très important de noter qu’il peut vous aider à gérer les cookies et les sessions, ainsi que l’authentification et la mise en commun automatique des connexions, entre autres choses. Son utilisation est gratuite et les développeurs Python l’utilisent pour télécharger des pages avant d’utiliser un analyseur syntaxique pour extraire les données requises.

## BeautifulSoup

BeautifulSoup facilite le processus d’analyse syntaxique des données contenues dans les pages Web. Il se place au-dessus d’un analyseur HTML ou XML et vous offre des méthodes Python permettant d’accéder aux données. BeautifulSoup est devenu l’un des outils de scraping Web les plus importants du marché en raison de la facilité d’analyse syntaxique qu’il offre. En fait, la plupart des didacticiels de scraping Web utilisent BeautifulSoup pour enseigner aux débutants comment écrire des scrapeurs Web. Lorsqu’il est utilisé avec Requests pour envoyer des requêtes HTTP, les web scrapers deviennent plus faciles à développer – beaucoup plus faciles qu’avec Scrapy ou PySpider. Selenuim Scrapy, Requests et BeautifulSoup ne vous aideront pas si un site Web est Ajaxifié, c’est-à-dire s’il dépend de requêtes AJAX pour charger certaines parties d’une page via JavaScript. Si vous accédez à une telle page, vous devez utiliser Selenium, qui est un outil d’automatisation du navigateur Web. Il peut être utilisé pour automatiser les navigateurs sans tête tels que Chrome et Firefox sans tête. Les versions plus anciennes peuvent automatiser PhantomJS. Maintenant que nous avons fait le tour de la théorie sur le webscraping, passons à la pratique avec un cas d’exemple. Pour une meilleure compréhension de ce cas d’exemple vous devez avoir une connaissance de base sur le langage de programmation python. Nous avons dédié tout un article sur ce sujet, comme nous l’avons mentionné au tout début de cette chronique. Et si vous souhaitez apprendre un autre langage fortement utilisé dans le milieu du Big Data, nous vous offrons cette formation sur la programmation Scala.

## 3- Les algorithmes utilisés

1. Algorithme d’extraction des données à partir de la page **web**https://www.worldometers.info/coronavirus

#La page Web que nous allons ouvrir se trouve à l'URL suivante

page = requests.get("https://www.worldometers.info/coronavirus")

page.status\_code

page.content

#importer la bibliothèque BeautifulSoup

from bs4 import BeautifulSoup

#Instauration de la classe BeautifulSoup

soup = BeautifulSoup(page.content, 'lxml')

#Où soupe est une variable contenant le code HTML de la page Web

#Formatons-le bien, en utilisant la méthode prettify comme contraste comme cellule 3

print(soup.prettify())

#Rechercher la table à extraire

table = soup.find('table', attrs={'id': 'main\_table\_countries\_today'})

table

La première partie du code nous génère le code html source de la page web et le stock dans la variable page. Ensuite, on le stock dans la variable soup en utilisant la fonction BeautifulSoup de la bibliothèque bs4.Finalement, on cherche la table à extraire dont les attributs sont {‘id’,’main\_table\_countries\_today’)

1. **La transformation de la table extraie en data Frame**

rows = table.find\_all("tr", attrs={"style": ""})  # Extraction du texte HTML qui represente la table

data = []

for i,item in enumerate(rows):

    if i == 0:

        data.append(item.text.replace(",","").strip().split("\n")[:13])    # La fonction text.strip().split("\n") transform le texte en une liste des chaines de caractères

    else:                                                  #L'insertion ligne par ligne dans data

        data.append(item.text.replace(",","").strip().split("\n")[:12])

data

!pip install dask

!pip install "dask[dataframe]"

import warnings

warnings.simplefilter(action='ignore', category=FutureWarning)

import pandas as pd

import dask.dataframe as dd

dt = pd.DataFrame(data)          # Transformer data sous format dataframe

dt = pd.DataFrame(data[1:], columns=data[0][:12]) #les lignes sont data[1:] et les colonnes sont le  header data[0][:12]

dt.head()

1. **Transformation des données d’objet à float**

# Transformation des données d'objet à float

data\_covid["TotalCases"] = pd.to\_numeric(data\_covid.TotalCases, errors='coerce')

data\_covid["NewCases"] = pd.to\_numeric(data\_covid.NewCases, errors='coerce')

data\_covid["TotalDeaths"] = pd.to\_numeric(data\_covid.TotalDeaths, errors='coerce')

data\_covid["NewDeaths"] = pd.to\_numeric(data\_covid.NewDeaths, errors='coerce')

data\_covid["TotalRecovered"] = pd.to\_numeric(data\_covid.TotalRecovered, errors='coerce')

data\_covid["NewRecovered"] = pd.to\_numeric(data\_covid.NewRecovered, errors='coerce')

data\_covid["ActiveCases"] = pd.to\_numeric(data\_covid.ActiveCases, errors='coerce')

data\_covid["SeriousCritical"] = pd.to\_numeric(data\_covid.SeriousCritical, errors='coerce')

1. **Afficher les dix premier pays les plus touchés par covid19 dans le monde**

#Visualiser les pays les plus touchés

figure=go.FigureWidget(layout=go.Layout())

def highlight\_col(x):# Changer les couleurs des colonnes

  r='background-color:green'

  y='background-color:purple'

  g='background-color:yellow'

  p='background-color:gray'

  df1=pd.DataFrame('',index=x.index,columns=x.columns)

  df1.iloc[:,2]=y

  df1.iloc[:,4]=r

  df1.iloc[:,6]=g

  df1.iloc[:,1]=p

  return df1

def show\_cases(n):

  n=int(n)

  return data\_covid.sort\_values('TotalCases',ascending=False).head(n).style.apply(highlight\_col,axis=None) #Afficher les 10 premiers pays touchés par covid19

interact(show\_cases,n='10')

ipywLayout=widgets.Layout(border='solid 2px green')

ipywLayout.display='none'

widgets.VBox([figure],layout=ipywLayout)

1. **Visualisation des 30 pays les plus touchés**

covid\_class=data\_covid.sort\_values('TotalCases',ascending=False)

px.bar(

    covid\_class.head(30),

    x='CountryOther',

    y='TotalCases',

    title='Les 30 pays les plus touchés par covid19',

    color\_discrete\_sequence=['pink'],

    height=700,

    width=800

)

1. **Visualisation des 30 pays ayant le grand nombre de décès**

covid\_class=data\_covid.sort\_values('TotalCases',ascending=False)

px.bar(

    covid\_class.head(30), # les 30 premiers pays par rapport au nombre de décès

    x='CountryOther',

    y='TotalDeaths',

    title='Les 30 pays ayant plus de décès',

    color\_discrete\_sequence=['green'],

    height=700,

    width=800

)

1. **Visualisation des 30 pays ayant le plus grand nombre des cas guérissent**

covid\_class=data\_covid.sort\_values('TotalCases',ascending=False)

px.bar(

    covid\_class.head(30), # les 30 premiers pays par rapport au nombre de décès

    x='CountryOther',

    y='TotalRecovered',

    title='Les 30 pays ayant plus d états guérées',

    color\_discrete\_sequence=['red'],

    height=700,

    width=800

)

1. **Visualisation des 30 pays ayant le grand nombre des cas actifs**

covid\_class=data\_covid.sort\_values('TotalCases',ascending=False)

px.bar(

    covid\_class.head(30), # les 30 premiers pays par rapport au nombre des cas total

    x='CountryOther',

    y='ActiveCases',

    title='Les 30 pays ayant plus des cas actives',

    color\_discrete\_sequence=['grey'],

    height=700,

    width=800

)

1. **Ajouter au Dataset les informations : Population, Longitude et Latitude**

# Lecture de la base de dataset de la population/pays

data=pd.read\_csv("/content/covid\_donness.csv")

population=pd.read\_csv("/content/data\_poplulation (1) (1).csv")

# Ajout de la colonne "population"

data["population"] = [0 for i in range(222)]

population

# Remplissage de  la colonne "Population"

for i in range(222):

  for j in range(234):

    if data['CountryOther'][i]==population['Country '][j]:

       data['population'][i]=population['Population'][j]

# Lecture de la dataset de localisation (latitude et longitude de chaque pays)

location=pd.read\_csv("/content/World latitudes and longitudes1.csv")

location

# Ajout des colonnes "Latitude" et "Longitude"

data["latitude"] = [0 for i in range(222)]

data["longitude"] = [0 for i in range(222)]

1. **Visualiser les données sur une carte**

import folium

import numpy as np

# Visualisation des données sur une carte

m=folium.Map(location=[31.7917,-7.0926],zoom\_start=6, max\_zoom=12,min\_zoom=2)

for i in range(0,len(data)):

  folium.Circle(

      location=[data.iloc[i]['latitude'],data.iloc[i]['longitude']],

      fill=True,

      radius=(int((np.log(data.iloc[i,14]+1.00001)))+0.2)\*25000,

      color='blue',

      fill\_color='blue',

      tooltip = "<div style='margin: 0; background-color: pink; color: blue;'>"+

                    "<h4 style='text-align:center;font-weight: bold'>"+data.iloc[i]['CountryOther'] + "</h4>"

                    "<hr style='margin:10px;color: white;'>"+

                    "<ul style='color: black;;list-style-type:circle;align-item:left;padding-left:20px;padding-right:20px'>"+

                        "<li>Population: "+str(data.iloc[i,14])+"</li>"+

                        "<li> TotalCases:   "+str(data.iloc[i,4])+"</li>"+

                        "<li>Les cas actives: "+ str(np.round(data.iloc[i,9]/(data.iloc[i,-1]+1.00001)\*100,2))+ "</li>"+

                    "</ul></div>",

        ).add\_to(m)

m

Représenter chaque pays sur la carte en utilisant des cercles dont le rayon est en fonction de la population dans ce pays.

# Visualisation des données sur une carte

m=folium.Map(location=[31.7917,-7.0926],zoom\_start=6, max\_zoom=12,min\_zoom=2)

for i in range(0,len(data)):

  folium.Circle(

      location=[data.iloc[i]['latitude'],data.iloc[i]['longitude']],

      fill=True,

      radius=(int((np.log(data.iloc[i,4]+1.00001)))+0.2)\*35000,

      color='red',

      fill\_color='red',

      tooltip = "<div style='margin: 0; background-color: black; color: white;'>"+

                    "<h4 style='text-align:center;font-weight: bold'>"+data.iloc[i]['CountryOther'] + "</h4>"

                    "<hr style='margin:10px;color: white;'>"+

                    "<ul style='color: white;;list-style-type:circle;align-item:left;padding-left:20px;padding-right:20px'>"+

                        "<li>Confirme: "+str(data.iloc[i,4])+"</li>"+

                        "<li>Deces:   "+str(data.iloc[i,6])+"</li>"+

                    "</ul></div>",

        ).add\_to(m)

m

Représenter chaque pays sur la carte en utilisant des cercles dont le rayon est en fonction de nombre des cas total dans ce pays.

1. **Visualisation de la relation entre la population et le nombre total des cas pour chaque pays**

plt.rcParams["figure.figsize"] = (20,20)

g = sns.lmplot("population","TotalCases",data\_class.head(40),hue= "CountryOther",fit\_reg=False,height=10)

1. **Visualisation des cas guérisses et cas décès pour les 30 pays les plus touchés**

import plotly.graph\_objects as go

months = data\_class.head(30)['CountryOther']

fig = go.Figure()

fig.add\_trace(go.Bar(

    x=months,

    y=data\_class.head(30)['TotalDeaths'],

    name='Total Deaths',

    marker\_color='indianred'

))

fig.add\_trace(go.Bar(

    x=months,

    y=data\_class.head(30)['TotalRecovered'],

    name='Total Recovered',

    marker\_color='lightsalmon'

))

# Here we modify the tickangle of the xaxis, resulting in rotated labels.

fig.update\_layout(barmode='group', xaxis\_tickangle=-45)

fig.show()

1. **Visualisation des cas guérisses et cas décès pour des pays moins touchés.**

import plotly.graph\_objects as go

months = data\_class[140:200]['CountryOther']

fig = go.Figure()

fig.add\_trace(go.Bar(

    x=months,

    y=data\_class[140:200]['TotalDeaths'],

    name='Total Deaths',

    marker\_color='indianred'

))

fig.add\_trace(go.Bar(

    x=months,

    y=data\_class[140:200]['TotalRecovered'],

    name='Total Recovered',

    marker\_color='lightsalmon'

))

# Here we modify the tickangle of the xaxis, resulting in rotated labels.

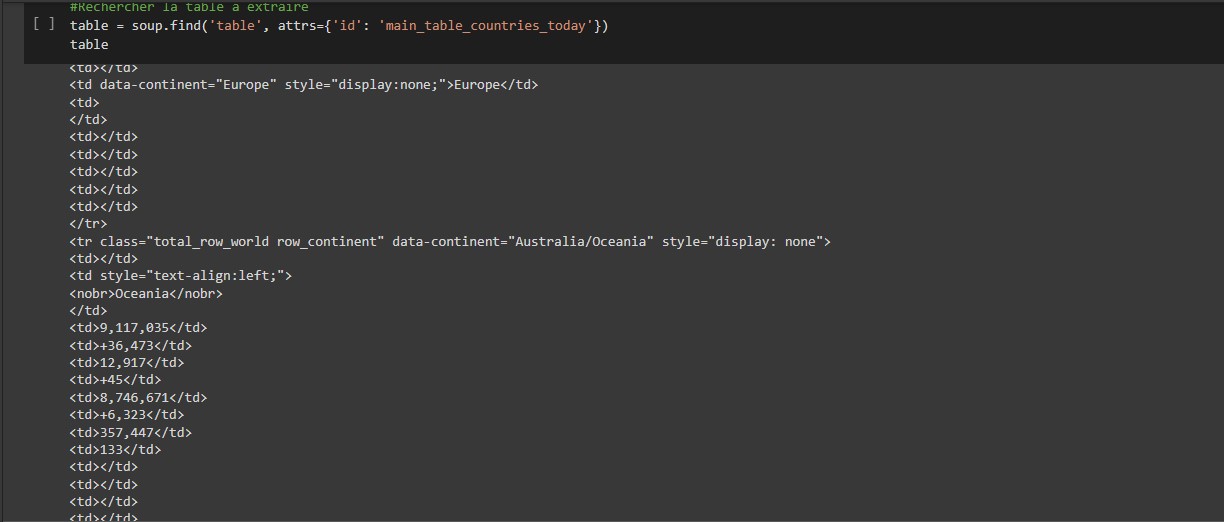
fig.update\_layout(barmode='group', xaxis\_tickangle=-45)

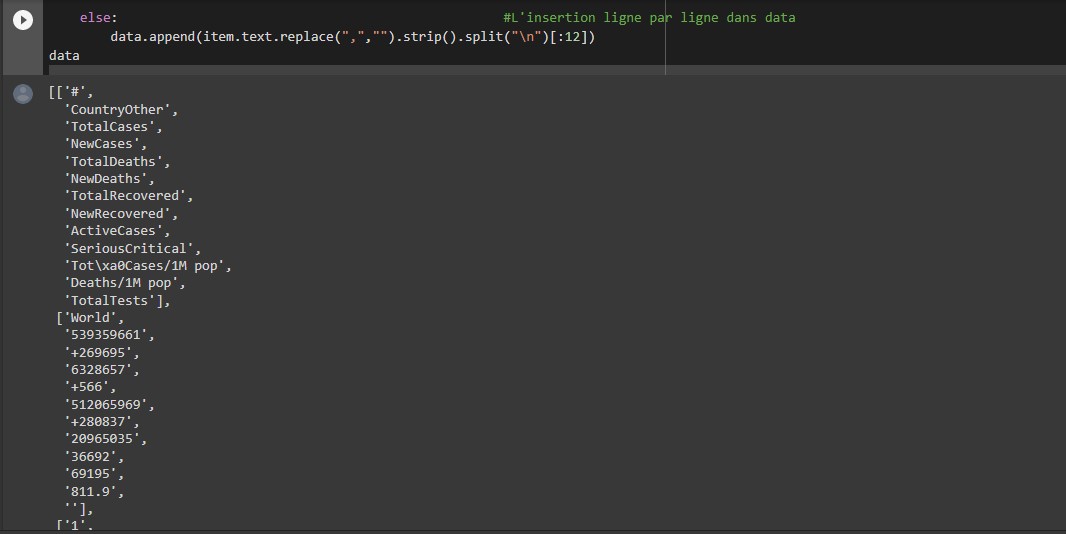
fig.show()

# II- Résultats

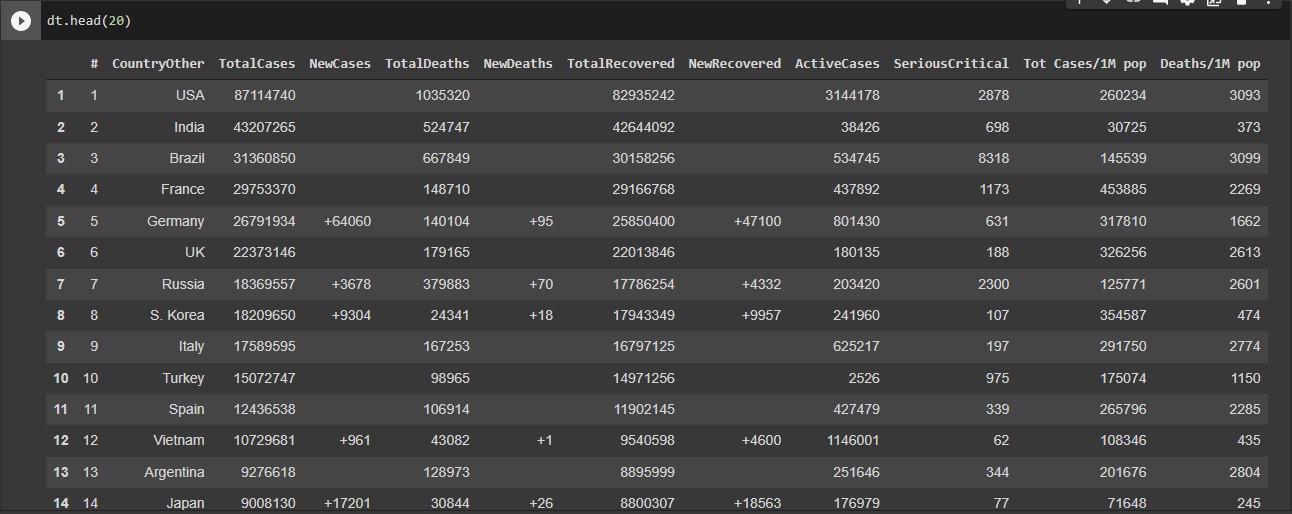
## Extraction des données

**Extraire les données et les enregistrer dans une liste**





1. Affichage de la base de dataset finale



## newplot (3).pngVisualisation

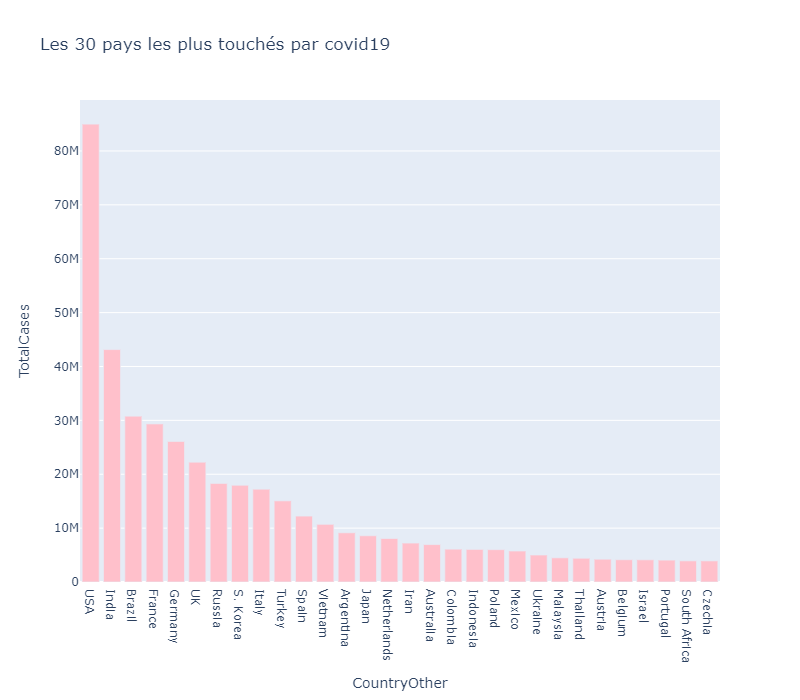
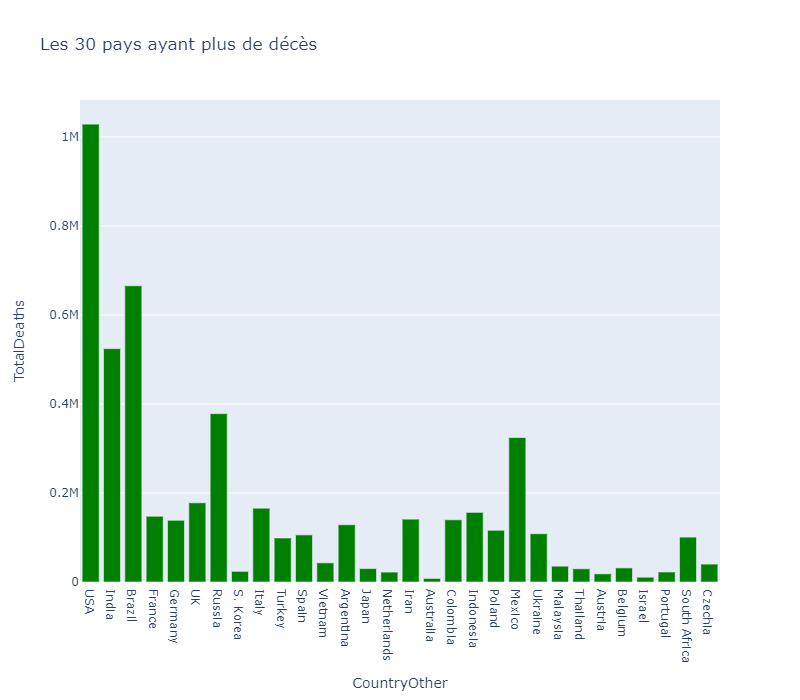


Figure 4 : Nombre des cas total. Figure 4 : Nombre des cas guérit.

* USA, Barazil, Inde et l’Allemand ont les plus grands nombres des cas touchés.
*  De même, USA, Barazil, Inde, l’Allemand et la France ont les plus grands nombres des cas guérissent.

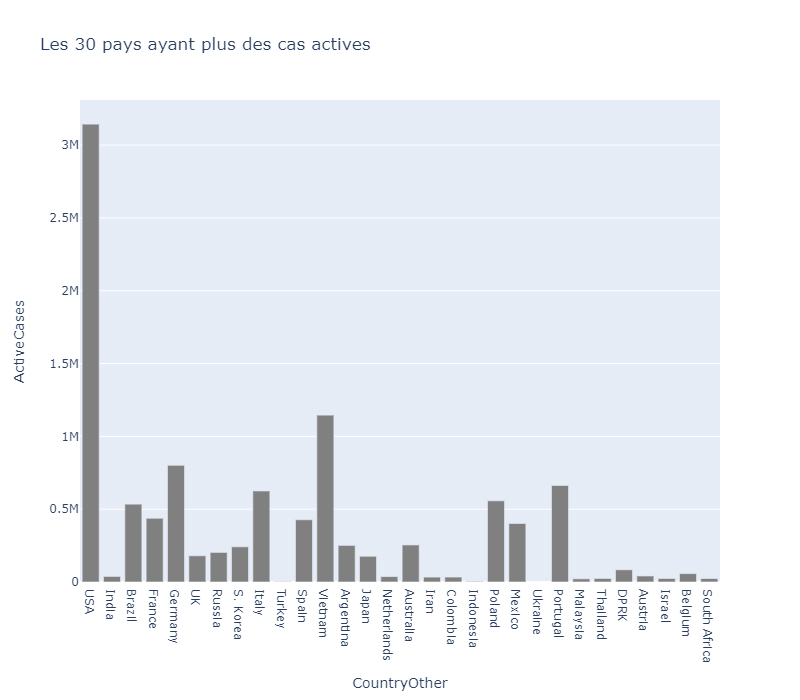


Figure 6 : Nombre des morts. Figure 7 : Nombre des cas actifs.

* USA, Barazil, Inde, Russe et la Mexique ont les plus grands nombre des morts.
* USA, Vietnam, Allemand, Portugal et Italie ont les plus grands nombre des cas actifs.

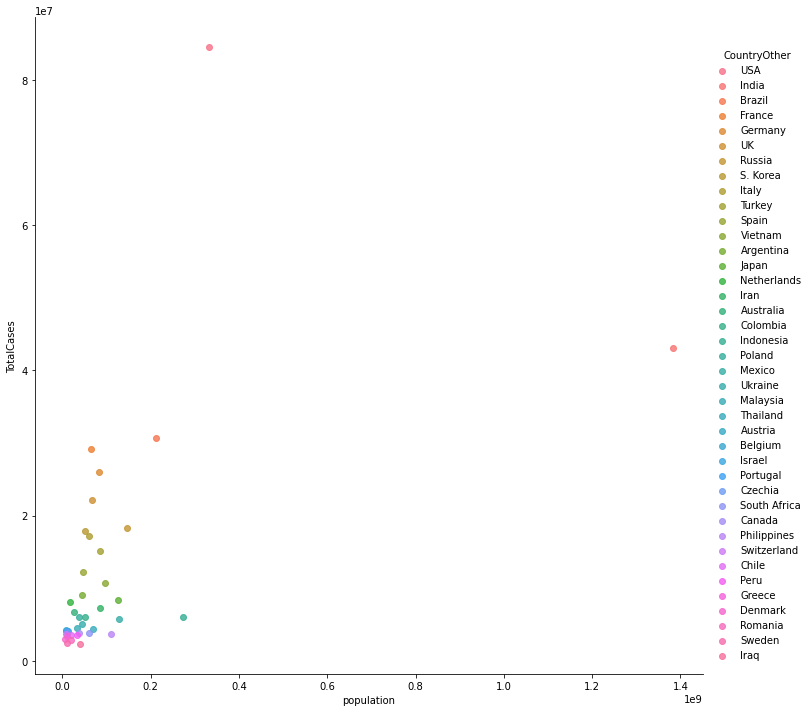
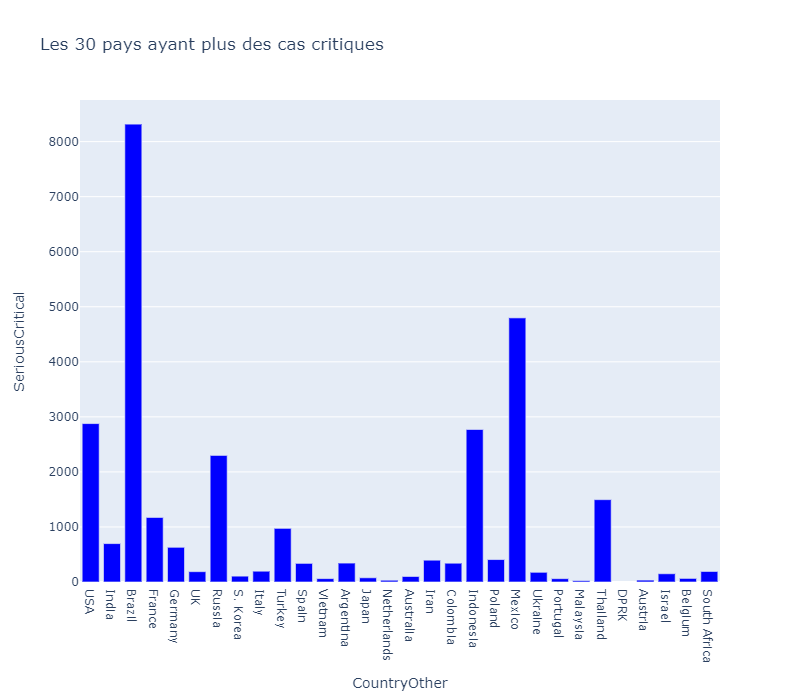


Figure 8 : Nombre des cas critiques. Figure 9 : Relation nombre des Cas-Population.

* Barazil, Mexique, USA, Indonésie et la Russe ont les plus grands nombres des cas critiques

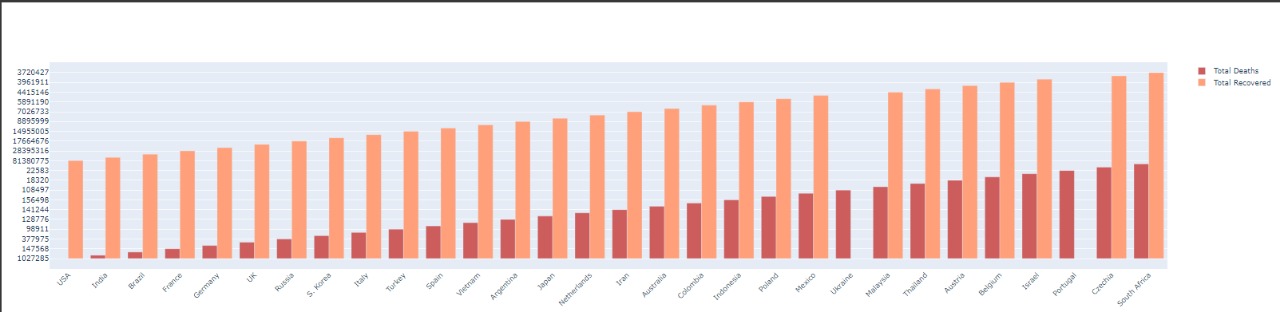


Figure 10 : comparaison entre nombre des cas guérissent et les morts (plus touchés).



 Figure 11 : comparaison entre nombre des cas guérissent et les morts (moins touchés).

Figure 12 : la densité de population pour chaque pays.

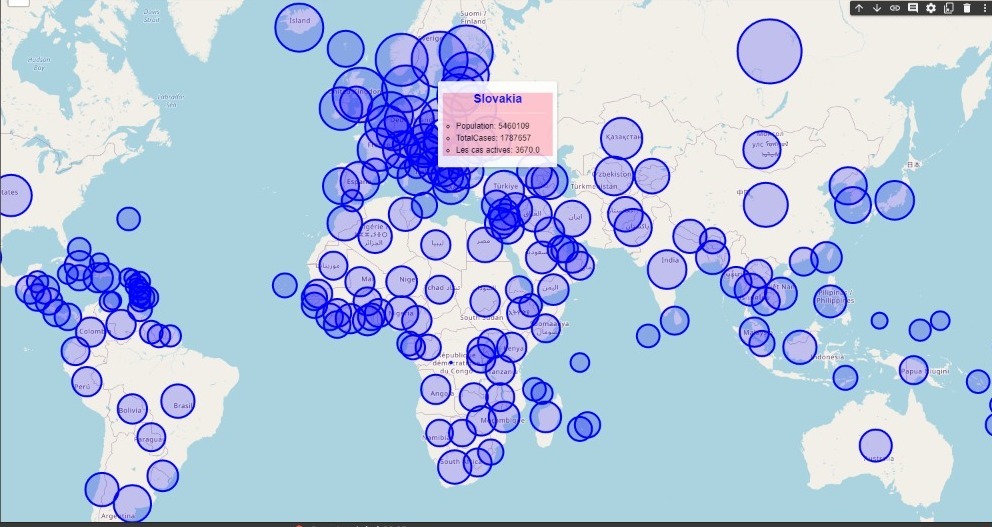


Figure 13 : Les cas touchés pour chaque pays.

# III-Analyse

## Les pays les plus touchés

* Les pays les plus touchés par covi19 se situent en : Amérique, Europe et Nord d'Assi .Par contre, en sud d'Afrique et sud d'Assi on remarque qu'ils sont moins touchés. **(Voir figure13)**

**Explication :**

* L'Amérique, l’Inde, La Russe et Assi ont une population importante par rapport aux autres régions. Toutefois, le sud d'Afrique se caractérise par une grande population comme même mais il n'est pas touché qu'avec la même degré que les régions cités avant ! **(Voir figure 12)**
* En effet, l’Europe, Nord d'Amérique et Nord de Assi ont un climat océanique ce qui fait, toujours l'atmosphère est froid ce qui aide le virus à survivre plus. **(Voir figure 13)**

## Relation entre population et cas contaminées – Raisons politiques et économiques

* On remarque que le nombre des cas dépend du volume de la population, par exemple on prend le cas usa de la population 331341050 ou on trouve 84473447 cas depuis le début de corona d'autre part on trouve Iraq ou le nombre des cas égal a 2327269 avec une population 40,22 millions. **(Voir figure 9)**
* Dans la plupart des cas, on constate que les pays développés ont un nombre des cas guérissent très important par rapport au nombre total des morts. Par exemple on prend le cas de USA ou on a total recorvered 81380775.0 et nombre total des morts 1027285 d'autre part on prend un exemple des pays sous-développés par exemple la South d’Afrique ou on trouve un total de des guérissent 3739543 et un total des morts 3739543 .C'est totalement logique parce que les pays développés ont les moyens tels que : les médicaments et les hôpitaux**. (Voir figures 11 et 10)**

# Conclusion

En guise de conclusion, le covid19 a fait une des journaux aux cours des dernières années. Cette épidémie a touchée un grand nombre de personnes et entrainé de nombreux décès.

Et pourtant, lorsque la covid-19 est arrivée, le monde n’était pas prêt. La pandémie a relevé des lacunes dans notre façon de penser et de mesurer la préparation aux épidémies. Certains pays ont étaient mis en difficulté malgré des systèmes de préparation plus solides selon les critères classiques, tandis que d’autres, dont les systèmes étaient jugés plus fragiles, ont su réagir avec vigueur. Et donc nous avons remarqués que le degré de ma contamination et l’effet sur chaque pays du monde n’était pas le même. L’objectif de ce projet est de ce basé sur les données fournit par la page web «https://www.worldometers.info/coronavirus » pour analyser les raisons derrière cette différence. D’abord la variation de la population pour chaque pays, Les pays les plus grands tels que USA, Inde…ont été plus affectés .Par contre, des pays comme

South d’Afrique n’avait pas beaucoup des cas contaminés. En plus, de la démographie, on trouve aussi la position géographique. Comme l’Europe à titre d’exemple a un climat océanique, l’atmosphère est toujours froide. Par contre, l’Afrique généralement a un climat sec et elle a moins des cas que l’Amérique ou l’Europe. Sans oublier, des raisons économiques, et la disponibilité des moyens nécessaires et des hôpitaux.

Des facteurs clés qui ne sont pas pris en compte par les indicateurs classiques utilisés pour mesurer le niveau de préparation aux épidémies, comme la prise de mesures politiques fortes et rapides, une bonne communication et une gouvernance de qualité, ont joué un rôle déterminant. Si la préparation est essentielle, l’inaction peut s’avérer mortelle.

Dans les moments de crise de santé publique, communiquer clairement et efficacement, tout en travaillant en collaboration avec les partenaires et les communautés, peut avoir un impact colossal sur l’issue de la riposte.

Dans l’ensemble, les pays qui ont le mieux réussi à maîtriser la COVID-19 étaient à la fois mieux préparés et disposaient d’une gouvernance solide tout au long de leur riposte à la pandémie. Il est essentiel de réunir ces deux conditions pour sauver des vies.

**Bibliothèque**

<https://www.oecd.org/coronavirus/policy-responses/l-impact-territorial-du-covid-19-gerer-la-crise-entre-niveaux-de-gouvernement-2596466b/>

<https://www.lapresse.ca/debats/opinions/2020-06-07/les-impacts-sociaux-de-la-covid-19-place-a-la-solidarite-d-apres-crise>

<https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.fondapol.org%2Fetude%2Fpauvrete-dans-le-monde-une-baisse-menacee-par-la-crise-sanitaire%2F&psig=AOvVaw3ui5pK6ZoRKGbWg-nc7SKP&ust=1654098052192000&source=images&cd=vfe&ved=0CAwQjRxqFwoTCNi9hIalivgCFQAAAAAdAAAAABAD>

<https://www.data-transitionnumerique.com/web-scraping-python/>

<https://fr.wikipedia.org/wiki/Web_scraping>

<https://webscraper.io/blog/brief-history-of-web-scraping>