

# Introduction à Gin

[Repo github des exercices](#)

## Présentation de Gin

[Gin](#) est un framework web HTTP écrit en Go.

Il dispose d'une API de type [Martini](#), mais avec des performances jusqu'à 40 fois plus rapides que Martini. Si vous avez besoin de performances époustouflantes, procurez-vous du Gin (le framework hein !).

Gin simplifie de nombreuses tâches de codage associées à la création d'applications Web, y compris les services Web.

## Fonctionnalités

[Documentation Gin](#)

- Rapide
- Prise en charge des middlewares
  - Exemple : Logger, Authorization, GZIP ...
- Pas de crash
  - Possède un catcheur `panic` interne pour intercepter les erreurs et empêcher l'arrêt de notre API
- JSON Validation
- Gestion d'erreur
  - On peut gérer manuellement les erreurs interceptées

## Gin VS Node

Gin est INCROYABLEMENT rapide comparé à des concurrents de tous les jours.

Le fait qu'il soit codé en Go permet d'obtenir un binaire complet pesant ~10Mo et contenant notre serveur API au complet. Comparé à Node qui doit inclure toutes ses librairies, c'est beaucoup moins !

Untitled

Untitled

# Docker

On peut créer des images docker de notre API gin ULTRA légères, puisque le binaire est standalone, une image alpine suffit

```
ARG GO_VERSION=1.18

FROM golang:${GO_VERSION}-alpine AS builder

RUN apk update && apk add --no-cache alpine-sdk git

WORKDIR /api

COPY go.mod .
COPY go.sum .
RUN go mod download

COPY . .
RUN go build -o ./app ./main.go

FROM alpine:latest

RUN apk update && apk add --no-cache ca-certificates

WORKDIR /api
COPY --from=builder /api/app .

EXPOSE 8080

ENTRYPOINT ["/app"]
```

On peut faire plus simple, mais je vous montre une version très optimisée d'un Dockerfile pour faire tourner une app go dans un environnement ultra léger.

---

Revision #1

Created 2022-05-09 19:22:28 UTC by Noé Larrieu-Lacoste

Updated 2022-05-09 19:23:24 UTC by Noé Larrieu-Lacoste