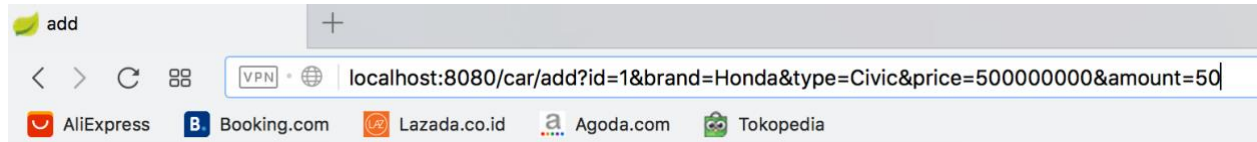


TUTORIAL

Pertanyaan 1: apakah hasilnya? Jika error, tuliskan penjelasan Anda.

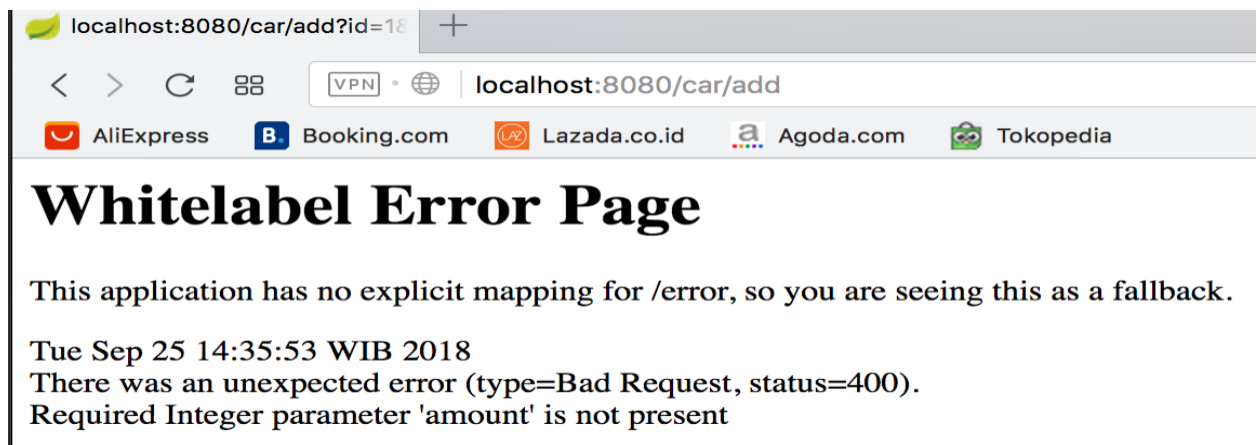
→ Hasilnya program berhasil dijalankan dan tidak menunjukkan *error*



Data Berhasil Ditambahkan

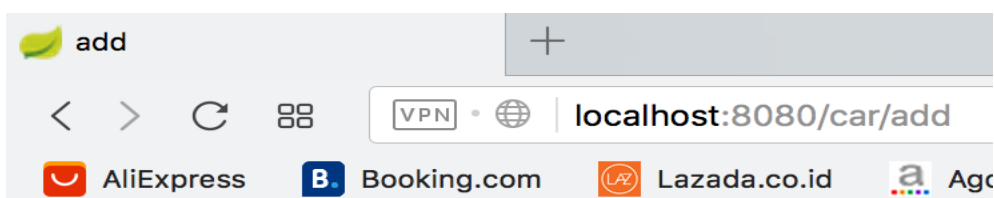
Pertanyaan 2: apakah hasilnya? Jika error, tuliskan penjelasan Anda.

→ Menunjukkan *Whitelabel Error Page*, dikarenakan parameter *amount* tidak terpenuhi dan memang parameter *amount* bersifat *required*. Sehingga, program tidak berhasil dijalankan.



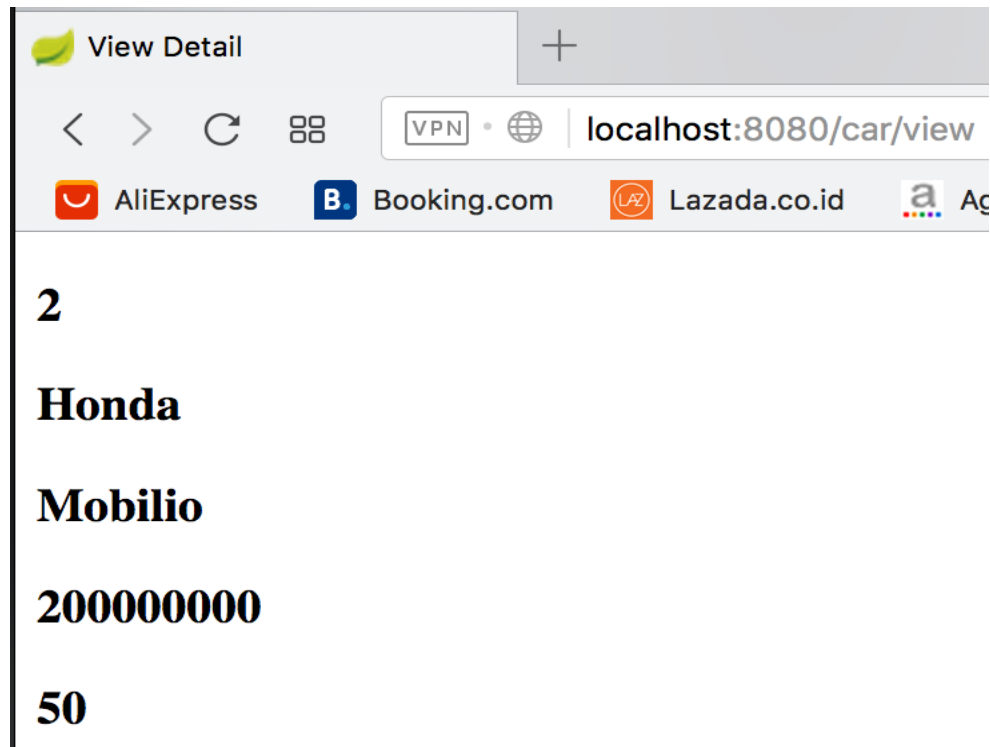
Pertanyaan 3: apakah data Car tersebut muncul? Jika tidak, mengapa?

→ Iya, data Car dapat dimunculkan. Dikarenakan pada saat proses “*add*” memang data *Car* berhasil ditambahkan, sehingga pada saat *user* menjalankan program “*view*” sesuai id yang diminta tidak memunculkan *error*.



Data Berhasil Ditambahkan

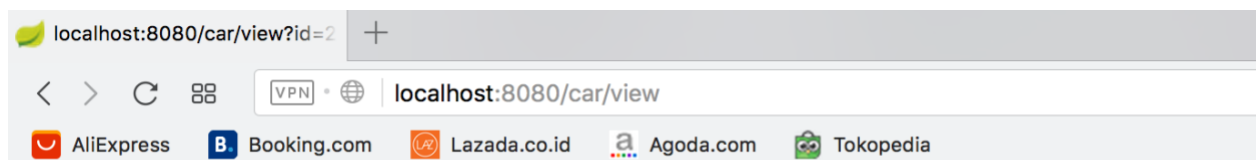
Tampilan Add



Tampilan view

Pertanyaan 4: apakah data Car tersebut muncul? Jika tidak, mengapa?

→ Menunjukkan *Whitelabel Error Page* dan data *Car* tidak muncul. Dikarenakan data *Car* merupakan *temporary variable* yang harus ditambahkan terlebih dahulu oleh *user* melalui program “add”, sehingga jika *user* tidak melalui program “add” data *Car* tidak akan muncul saat menjalankan program “view”.



Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

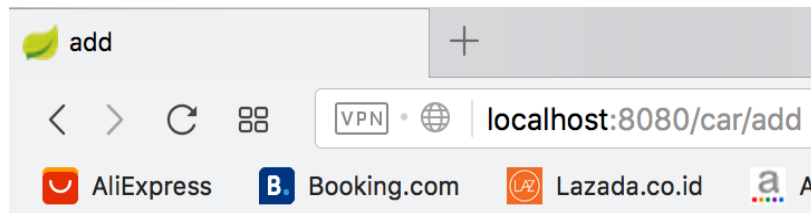
Tue Sep 25 14:07:44 WIB 2018

There was an unexpected error (type=Internal Server Error, status=500).

An error happened during template parsing (template: "class path resource [templates/view-car.html]")

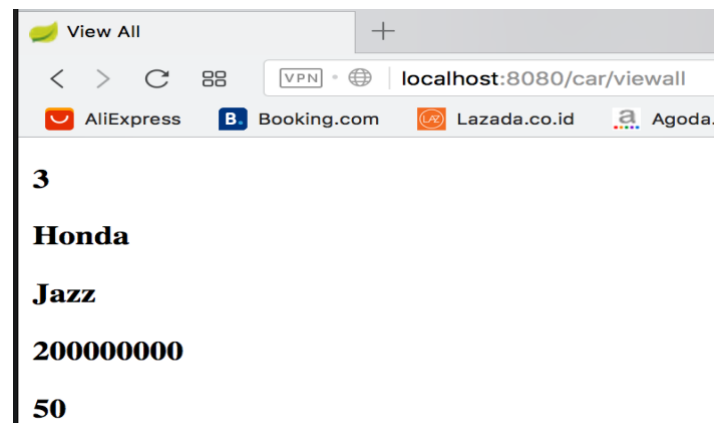
Pertanyaan 5: apakah data Car tersebut muncul?

→ Iya, semua data *Car* muncul. Dikarenakan pada saat proses “add” memang data *Car* berhasil ditambahkan, sehingga pada saat *user* menjalankan program “view all” tidak memunculkan *error*.



Data Berhasil Ditambahkan

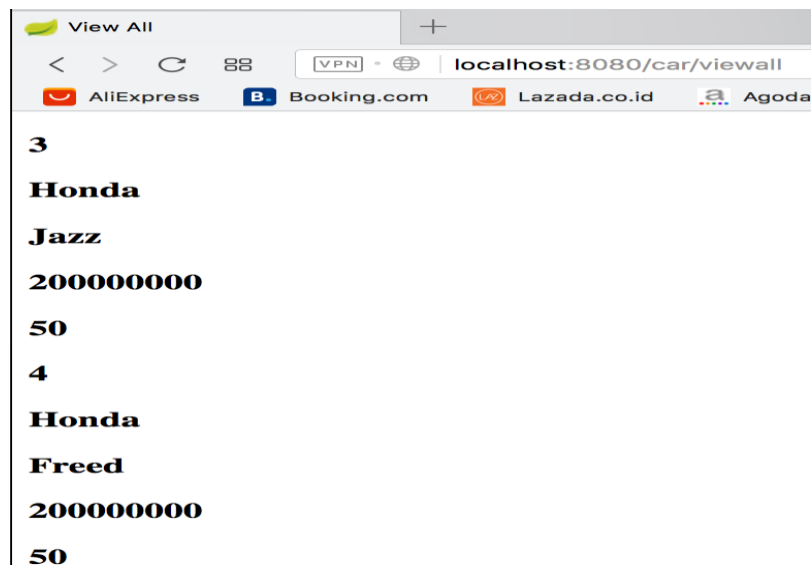
Tampilan Add



Tampilan View All

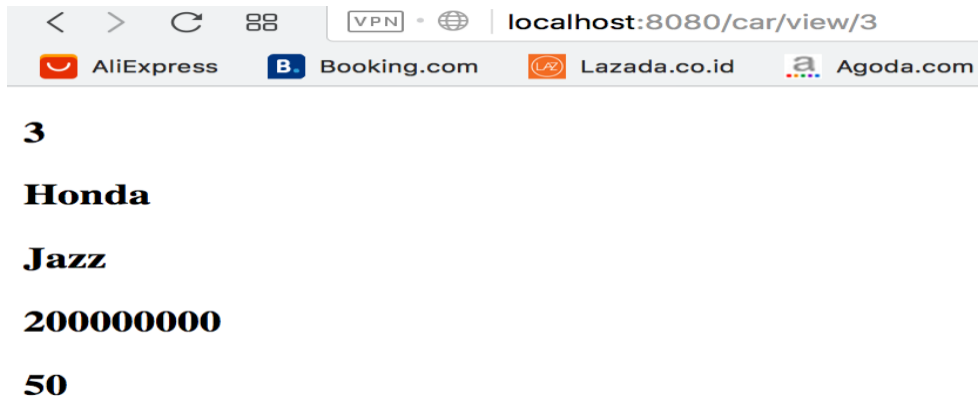
Pertanyaan 6: Apakah semua data Car muncul?

→ Iya, semua data Car muncul. Dikarenakan pada saat proses “add” memang data *Car* berhasil ditambahkan, sehingga pada saat *user* menjalankan program “view all” tidak memunculkan *error*.



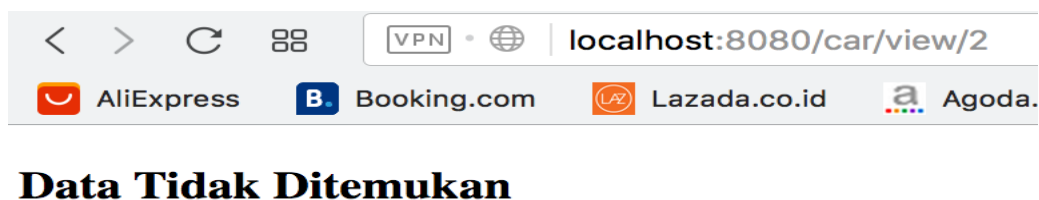
LATIHAN

1. Pada **CarController** tambahkan sebuah **method view** Cardengan menggunakan **Path Variable**. Misalnya, Anda ingin memasukkan data sebuah Car yang memiliki id 1, untuk melihat data yang baru dimasukkan tersebut dapat mengakses halaman **localhost:8080/car/view/1**.



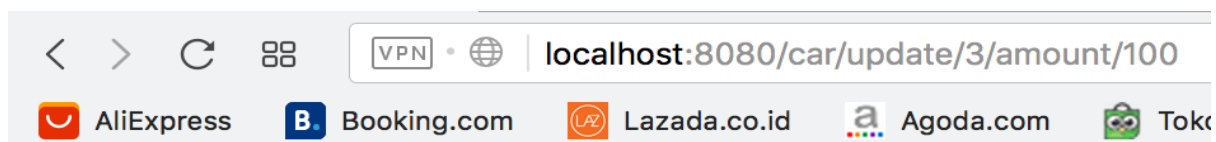
Tampilan View dengan ID = 3

Jika nomor id tidak diberikan atau tidak ditemukan kembalikan halaman *error* yang berisi informasi bahwa nomor id kosong atau tidak ditemukan.



Tampilan View dengan ID yang belum dimasukkan

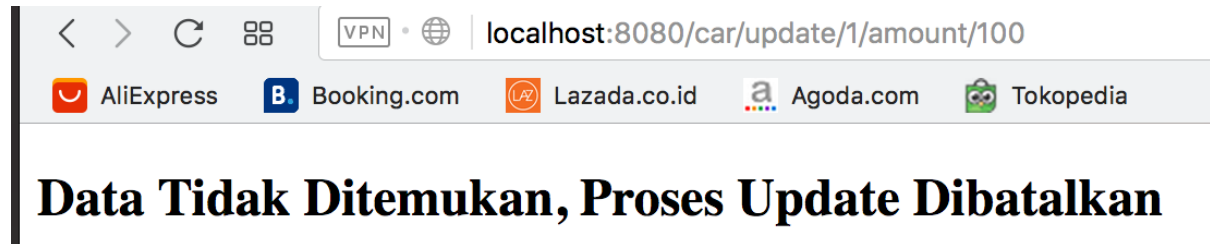
2. Tambahkan fitur untuk melakukan update amount dari Car berdasarkan id. Misalnya, setelah melakukan add Car pada soal nomor 1, cobalah untuk mengubah amount objek Car tersebut menjadi 100 dengan mengakses halaman **localhost:8080/car/update/1/amount/100**. Tampilkan sebuah halaman yang memberikan informasi bahwa data tersebut telah berhasil diubah.



Data Berhasil Di-update

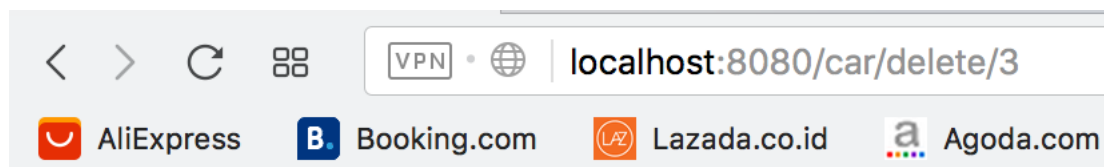
Tampilan Update dengan ID = 3 dan amount 100

Jika id tidak diberikan atau tidak ditemukan kembalikan halaman error yang berisi informasi bahwa id kosong atau tidak ditemukan dan proses *update* dibatalkan



Tampilan *Update* dengan ID yang belum dimasukkan

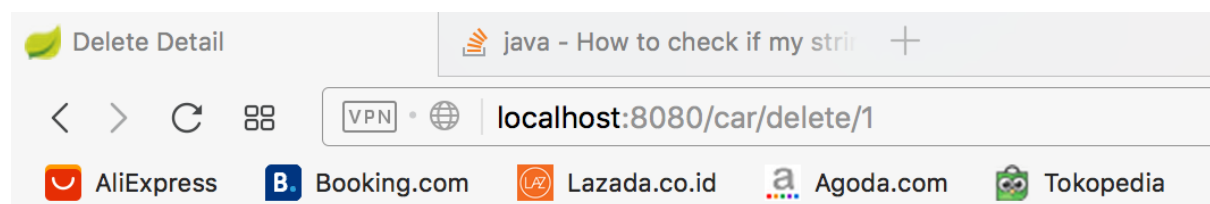
3. Tambahkan fitur untuk melakukan *delete* Car berdasarkan id. Misalnya, setelah melakukan perintah nomor 1 dan 2, cobalah untuk melakukan *delete* data tersebut dengan mengakses halaman **localhost:8080/car/delete/1**. Tampilkan sebuah halaman yang memberikan informasi bahwa data tersebut telah berhasil dihapus.



Data Berhasil Di-delete

Tampilan *delete* dengan ID = 3

Jika id tidak diberikan atau tidak ditemukan kembalikan halaman *error* yang berisi informasi bahwa id kosong atau tidak ditemukan dan proses *delete* dibatalkan



Data Tidak Ditemukan, Proses Delete Dibatalkan

Tampilan *delete* dengan ID yang belum dimasukkan