Adonis Framework Cheat Sheet

Adonis CLI (open a new terminal or a gitbash)

To create a new adonis project

```
adonis new web_app
```

to start the new web project

```
adonis serve --dev
```

to create a model

```
adonis make:model User
```

to create a new Controller

```
adonis make:controller UserController
```

Routing

to write a static route

```
Route.get('/home', async() => {
    return `Hello Adonis World`
})
```

passing parameters to the route

```
Route.get('/student/:name', async({ params }) => {
    return `Hello dear student: ${params.name}`
})
```

making the parameters optionals into a route

(only need to add the ? sign just after the city parameter)

```
Route.get('/country/:city?', async ({ params }) => {
    if (params.city) {
        return `The city: ${params.city} exists`
    }  else{
        return `unknown city`
    }
})
```

Linking a route to a controller and it¿s method

```
Route.post('/save-student', 'StudentController.store')
```

Structure of a resource controller

```
Route.resource('/users', 'UsersController')
```

Lucid ORM

fecth all records from a table

```
const users = await Users.all()
```

find a record by it’s id

```
const user = await User.find(params.id)
```

delete a single record

```
const deleted_user = await User.find(params.id)
await deleted_user.delete()
```

create a new user record

```
const new_user = new User()

new_user.name = request.input('name')
new_user.email = request.input('email')
new_user.phone = request.input('phone')

new_user.save()
```

usage of clausule where

```
const specified_user = await User.query().where('status', 1).
fetch()
```

throw and exception if a record is not finded

```
const user = User.findOrFail(params.id)
```

Models

If you wanna work with the users table, then

```
const Model = use('Model')

class User extends Model {
    static get table() {
        return 'users_table'
    }
}

module.exports = User
```

Views

Adonis has its own template engine called **Edge**

to show data into an edge file you can do this

```
<ul>
@each(dato in data)
    <li>{{ dato.name }}</li>
@endeach
</ul>
```

If you wanna add CSRF protection to you html forms, then

```
{{ csrfField() }}
```

Schema

to create a new table, you will make use of migrations, of the following mode

```
class categoriesSchema extends Schema {
    up () {
        this.create('categories', (table) => {
            table.increments()
            table.string('name', 30).notNullable().unique()
```

```
                table.boolean('status', 1).notNullable()
            })
        }
    }
```

Cheat Sheet created and maintained by:

Ing. Alfredo Paz

@IngAlfredoPaz

> *Only the effort pays off, please ever be honorable*