# Carity FE Fiels Tool

1. Environment

Options: [query path]? True/False

identifier.def.ts: generate default identifier for that form

route-path.def.ts: route path and query route path(if [query path] option is true)

environment.service.ts: generate base on identifier.def.ts&route-path.def.ts

1. Router:

Options: [has list]? True/False; [has profile]? True/False; [has add]? True/False

router-navigate.service.ts:

router.service.ts: if any view(list/profile/add) is true then don’t comment out the related code

1. Add

Options: [service api]?choose the specific api file, get the method and create mode

add-form-stepper-config.service.ts: update the message according to module name(path)

add.module.ts

add.service.ts: find model and creation method from service api file and generate this

1. List

Options: [list mode]? manual/config; [preview mode]? manual/remote/config

list.component.html: we have two types of table and three types of preview, generate them according to options

list.component.ts: columnConfigs(config) or displayedColumns(manual)

list.module.ts:

list.service.ts:

1. Panel

Options: [plural]? True/False; [has form]? True/False; [property]? property name

Folder name is “property-panel”

property-panel.component.html: generate this base on options

property-panel.component.ts

1. Panel Form:

Options: [flat]? True/False; [model]? if flat is true then we need to configure this

xxx-form.component.html: if flat is true then list all properties from model

xxx-form.component.ts: add empty rules

1. Profile

Options: [index model]? select index model file

[service api]? select service api file

[has attachment...]? True/False

panels: generate panels base on index model

profile-component.service.ts: generate this base on index model(generate panels first)

profile.component.html:

profile.component.ts: has attachment?

profile.module.ts:

profile.service.ts: generate this base on service api and index model

1. Query:

Options: [name]? the name of query; [has filter]? True/False

name.component.html: generate this according to [has filter]

name.component.ts: generate according to [has filter], generate an empty columnConfigs

name.module.ts:

name.service.ts: according to [has filter]

1. Query Root:

Options: [query service api]? choose the query service api file;

Generate queries base on [query service api]

query-module.service.ts: generate children base on [query service api] and queries

query.module.ts: generate base on queries and [query service api]

1. Services:

Combine environment and Router

1. Root Module:

Combine environment,router,add,list,profile,query

1. TODO...

Options input:

