/\*\* 为实体类型创建EntityCollectionService的核心元素. \*/

@Injectable()

export class EntityCollectionServiceElementsFactory {

**constructor**(

private entityDispatcherFactory: EntityDispatcherFactory,

private entityDefinitionService: EntityDefinitionService,

private entitySelectorsFactory: EntitySelectorsFactory,

private entitySelectors$Factory: EntitySelectors$Factory

) {}

/\*\*

\*获取为此实体类型创建EntityCollectionService的成分

\* @param entityName - name of the entity type

\*/

**create**<T, S$ extends EntitySelectors$<T> = EntitySelectors$<T>>(

entityName: string

): EntityCollectionServiceElements<T, S$> {

entityName = entityName.trim();

const definition = this.entityDefinitionService.getDefinition<T>(

entityName

);

const dispatcher = this.entityDispatcherFactory.create<T>(

entityName,

definition.selectId,

definition.entityDispatcherOptions

);

const selectors = this.entitySelectorsFactory.create<T>(

definition.metadata

);

const selectors$ = this.entitySelectors$Factory.create<T, S$>(

entityName,

selectors

);

return {

dispatcher,

entityName,

selectors,

selectors$,

};

}

}