DATOS INFORMATIVOS:

CARRERA: Desarrollo de Software	**PERIODO LECTIVO:** *2022 - 2023*
MATERIA: Desarrollo Web	**JORNADA: Matutina**
PERÍODO ACADÉMICO: Cuarto	**PARALELO:** A
NOMBRE: KEVIN SANI	
FECHA: 2023-01-31	

DTO
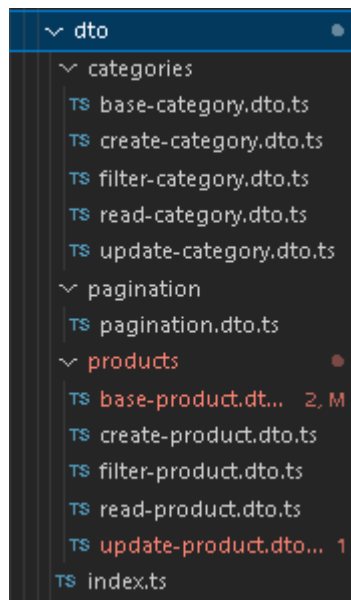
```
COMMIT_EDITMSG     TS base-product.dto.ts 2, M     TS pagination.dto.ts 1  ×

server > src > modules > ventas > dto > pagination > TS pagination.dto.ts > ...
 1   import { IsOptional, IsPositive, IsString, Min } from "class-validator";
 2
 3   export class PaginationDto{
 4       @IsOptional()
 5       @IsPositive()
 6       limit:number;
 7
 8       @IsOptional()
 9       @Min(0)
10       page:number;
11
12       @IsOptional()
13       @IsString()
14       search:string;
15
16       static gettOffset(limit:number, page:number):number{
17           return page * limit;
18       }
19
20   }
```

```
COMMIT_EDITMSG     TS base-product.dto.ts 2, M  ×     TS pagination.dto.ts 1

server > src > modules > ventas > dto > products > TS base-product.dto.ts > ⁑ BaseProductDto > 𝒫 title
 1   import {
 2       IsString,
 3       Allow,
 4       IsBoolean,
 5       IsNumber,
 6       IsNotEmpty,
 7       ArrayNotEmpty,
 8       IsArray,
 9       IsPositive,
10       IsInt,
11   } from 'class-validator';
12
13   import {
14       isBooleanValidationOptions,
15       isNotEmptyValidationOptions,
16       isNumberValidationOptions,
17       isStringValidationOptions,
18       IsArrayValidationOptions,
19       isPositiveValidationOptions,
20       IsIntValidationOptions
21   } from '@shared/validation';
22
23   //validar los campos de la clase
24
25   export class BaseProductDto {
26
27
28
29       @IsNotEmpty(isNotEmptyValidationOptions())
30       @IsString(isStringValidationOptions())
31       readonly title: string;
32
33
34       //shared validacion option para observar los mensajes de forma clara
35       @IsNotEmpty(isNotEmptyValidationOptions())
36       @IsNumber(isNumberValidationOptions())
37       @IsPositive(isPositiveValidationOptions)
38       readonly price: number;
39
40       @IsNotEmpty(isNotEmptyValidationOptions())
41       @IsString(isStringValidationOptions())
42       readonly description: string;
43
44       @ArrayNotEmpty(isNotEmptyValidationOptions())
45       @IsArray(IsArrayValidationOptions())
46       readonly images:string[];
47
48       @IsNotEmpty(isNotEmptyValidationOptions())
49       @IsInt(IsIntValidationOptions())
50       @IsString(isNumberValidationOptions())
51       readonly categpryId: number;
52
53
54
55
56   }
```

```typescript
COMMIT_EDITMSG      TS base-product.dto.ts 2, M      TS create-product.dto.ts 2 ×

server > src > modules > ventas > dto > products > TS create-product.dto.ts > ...
   1    /* eslint-disable prettier/prettier */
   2  ⌄ import { IsInt, IsPositive } from "class-validator";
   3    import { BaseProductDto } from "./base-product.dto";
   4    import {
   5        isBooleanValidationOptions,
   6        isNotEmptyValidationOptions,
   7        isNumberValidationOptions,
   8        isStringValidationOptions,
   9        IsArrayValidationOptions,
  10        isPositiveValidationOptions,
  11        IsIntValidationOptions
  12
  13      } from '@shared/validation';
  14
  15    //extends accede a los atributos de la clase padre
  16  ⌄ export class CreateProductDto extends BaseProductDto{
  17
  18
  19
  20
  21
  22    }
```

```typescript
COMMIT_EDITMSG      TS base-product.dto.ts 2, M      TS filter-product.dto.ts 4 ×

server > src > modules > ventas > dto > products > TS filter-product.dto.ts > ...
   1  ⌄ import { IsNumber, IsOptional, IsString } from 'class-validator';
   2    import {PaginationDto} from 'core/dto';
   3    import {
   4        isBooleanValidationOptions,
   5        isNotEmptyValidationOptions,
   6        isNumberValidationOptions,
   7        isStringValidationOptions,
   8        IsArrayValidationOptions,
   9        isPositiveValidationOptions,
  10        IsIntValidationOptions
  11      } from '@shared/validation';
  12
  13    import { title } from 'process';
  14
  15    //visualizar mediante paginas el nombre,cedula
  16  ⌄ export class FilterProductDto extends PaginationDto{
  17    @IsOptional()
  18    @IsString(isStringValidationOptions())
  19    readonly title:string;
  20  ⌄ /* limit:number;
  21    page:number;
  22    search:string; */
  23    @IsOptional()
  24    @IsNumber(isNumberValidationOptions())
  25    readonly categoryId:number;
  26
  27
  28
  29
  30    @IsOptional()
  31    @IsString({message:'el campo nomnbre debe ser string'})
  32        limit: number;
  33        page: number;
  34
  35
  36    }
  37
```

```
◈ COMMIT_EDITMSG       TS base-product.dto.ts 2, M       TS read-product.dto.ts 1  ✕

server > src > modules > ventas > dto > products > TS read-product.dto.ts > ...
  1
  2   import { Exclude, Expose } from 'class-transformer';
  3   import { BaseProductDto } from './base-product.dto';
  4
  5   @Exclude()
  6   export class ReadProductDto extends BaseProductDto {
  7     //exponer el valor del atributo
  8     @Expose()
  9     readonly title;
 10
 11     @Expose()
 12     readonly description;
 13
 14     @Expose()
 15     readonly price;
 16
 17     @Expose()
 18     readonly categoryId;
 19
 20     @Expose()
 21     readonly images;
 22
 23   }
```

```
◈ COMMIT_EDITMSG       TS base-product.dto.ts 2, M       TS update-product.dto.ts 1  ✕

server > src > modules > ventas > dto > products > TS update-product.dto.ts > ...
  1   import { BaseProductDto } from "./base-product.dto";
  2   import {PartialType} from '@nestjs/swagger';
  3
  4
  5   export class UpdateProductDto extends PartialType(BaseProductDto){
  6
  7
  8
  9   }
```

```
◈ COMMIT_EDITMSG       TS base-product.dto.ts 2, M       TS index.ts       ✕

server > src > modules > ventas > dto > TS index.ts
  1   //Products
  2   export * from './products/base-product.dto';
  3   export * from './products/create-product.dto';
  4   export * from './products/filter-product.dto';
  5   export * from './products/read-product.dto';
  6   export * from './products/update-product.dto';
  7
  8
  9   //Category
 10   export * from './categories/base-category.dto';
 11   export * from './categories/create-category.dto';
 12   export * from './categories/filter-category.dto';
 13   export * from './categories/read-category.dto';
 14   export * from './categories/update-category.dto';
 15
```