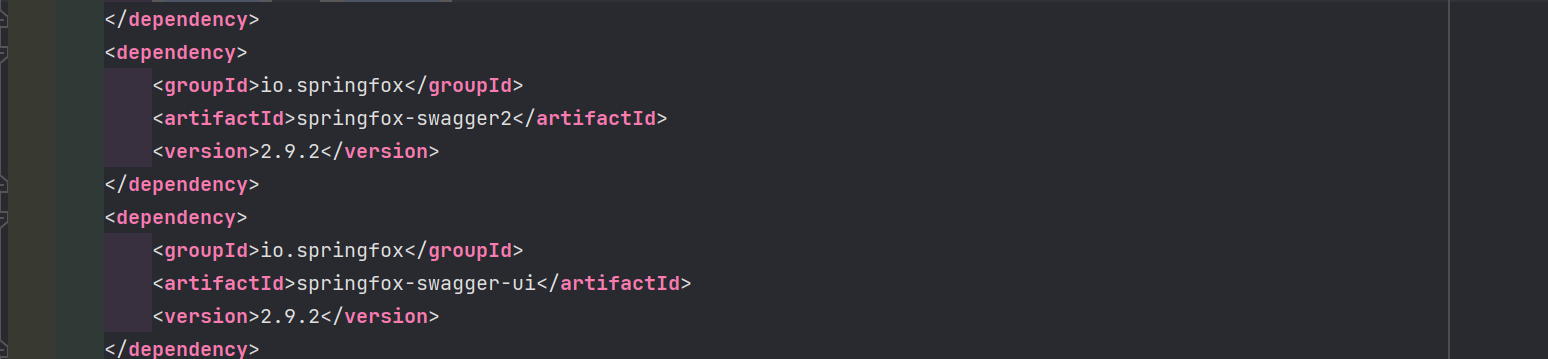
Name:

SaiMadhav

Task2

Genereating api using swagger code gen and latest and simple way using swagger2

I have added swagger and swagger ui dependency in to the pom of my project



Now I have annotated my project with @Enable swagger2

Text

Description automatically generated

And now I have hit the from post man to get the code generated automatically :

A screenshot of a computer

Description automatically generated with medium confidence

Code:

{

"swagger": "2.0",

"info": {

"description": "Api Documentation",

"version": "1.0",

"title": "Api Documentation",

"termsOfService": "urn:tos",

"contact": {},

"license": {

"name": "Apache 2.0",

"url": "http://www.apache.org/licenses/LICENSE-2.0"

}

},

"host": "localhost:8081",

"basePath": "/",

"tags": [

{

"name": "basic-error-controller",

"description": "Basic Error Controller"

},

{

"name": "task-controller",

"description": "Task Controller"

}

],

"paths": {

"/addServerDetails": {

"put": {

"tags": [

"task-controller"

],

"summary": "saveServers",

"operationId": "saveServersUsingPUT",

"consumes": [

"application/json"

],

"produces": [

"\*/\*"

],

"parameters": [

{

"in": "body",

"name": "task",

"description": "task",

"required": true,

"schema": {

"$ref": "#/definitions/Task"

}

}

],

"responses": {

"200": {

"description": "OK",

"schema": {

"type": "string"

}

},

"201": {

"description": "Created"

},

"401": {

"description": "Unauthorized"

},

"403": {

"description": "Forbidden"

},

"404": {

"description": "Not Found"

}

}

}

},

"/delete/{id}": {

"delete": {

"tags": [

"task-controller"

],

"summary": "deleteServer",

"operationId": "deleteServerUsingDELETE",

"produces": [

"\*/\*"

],

"parameters": [

{

"name": "id",

"in": "path",

"description": "id",

"required": true,

"type": "integer",

"format": "int32"

}

],

"responses": {

"200": {

"description": "OK",

"schema": {

"type": "string"

}

},

"204": {

"description": "No Content"

},

"401": {

"description": "Unauthorized"

},

"403": {

"description": "Forbidden"

}

}

}

},

"/error": {

"get": {

"tags": [

"basic-error-controller"

],

"summary": "error",

"operationId": "errorUsingGET",

"produces": [

"\*/\*"

],

"responses": {

"200": {

"description": "OK",

"schema": {

"type": "object",

"additionalProperties": {

"type": "object"

}

}

},

"401": {

"description": "Unauthorized"

},

"403": {

"description": "Forbidden"

},

"404": {

"description": "Not Found"

}

}

},

"head": {

"tags": [

"basic-error-controller"

],

"summary": "error",

"operationId": "errorUsingHEAD",

"consumes": [

"application/json"

],

"produces": [

"\*/\*"

],

"responses": {

"200": {

"description": "OK",

"schema": {

"type": "object",

"additionalProperties": {

"type": "object"

}

}

},

"204": {

"description": "No Content"

},

"401": {

"description": "Unauthorized"

},

"403": {

"description": "Forbidden"

}

}

},

"post": {

"tags": [

"basic-error-controller"

],

"summary": "error",

"operationId": "errorUsingPOST",

"consumes": [

"application/json"

],

"produces": [

"\*/\*"

],

"responses": {

"200": {

"description": "OK",

"schema": {

"type": "object",

"additionalProperties": {

"type": "object"

}

}

},

"201": {

"description": "Created"

},

"401": {

"description": "Unauthorized"

},

"403": {

"description": "Forbidden"

},

"404": {

"description": "Not Found"

}

}

},

"put": {

"tags": [

"basic-error-controller"

],

"summary": "error",

"operationId": "errorUsingPUT",

"consumes": [

"application/json"

],

"produces": [

"\*/\*"

],

"responses": {

"200": {

"description": "OK",

"schema": {

"type": "object",

"additionalProperties": {

"type": "object"

}

}

},

"201": {

"description": "Created"

},

"401": {

"description": "Unauthorized"

},

"403": {

"description": "Forbidden"

},

"404": {

"description": "Not Found"

}

}

},

"delete": {

"tags": [

"basic-error-controller"

],

"summary": "error",

"operationId": "errorUsingDELETE",

"produces": [

"\*/\*"

],

"responses": {

"200": {

"description": "OK",

"schema": {

"type": "object",

"additionalProperties": {

"type": "object"

}

}

},

"204": {

"description": "No Content"

},

"401": {

"description": "Unauthorized"

},

"403": {

"description": "Forbidden"

}

}

},

"options": {

"tags": [

"basic-error-controller"

],

"summary": "error",

"operationId": "errorUsingOPTIONS",

"consumes": [

"application/json"

],

"produces": [

"\*/\*"

],

"responses": {

"200": {

"description": "OK",

"schema": {

"type": "object",

"additionalProperties": {

"type": "object"

}

}

},

"204": {

"description": "No Content"

},

"401": {

"description": "Unauthorized"

},

"403": {

"description": "Forbidden"

}

}

},

"patch": {

"tags": [

"basic-error-controller"

],

"summary": "error",

"operationId": "errorUsingPATCH",

"consumes": [

"application/json"

],

"produces": [

"\*/\*"

],

"responses": {

"200": {

"description": "OK",

"schema": {

"type": "object",

"additionalProperties": {

"type": "object"

}

}

},

"204": {

"description": "No Content"

},

"401": {

"description": "Unauthorized"

},

"403": {

"description": "Forbidden"

}

}

}

},

"/findAllServers": {

"get": {

"tags": [

"task-controller"

],

"summary": "getServers",

"operationId": "getServersUsingGET",

"produces": [

"\*/\*"

],

"responses": {

"200": {

"description": "OK",

"schema": {

"type": "array",

"items": {

"$ref": "#/definitions/Task"

}

}

},

"401": {

"description": "Unauthorized"

},

"403": {

"description": "Forbidden"

},

"404": {

"description": "Not Found"

}

}

}

},

"/findAllServers/{id}": {

"get": {

"tags": [

"task-controller"

],

"summary": "getServer",

"operationId": "getServerUsingGET",

"produces": [

"\*/\*"

],

"parameters": [

{

"name": "id",

"in": "path",

"description": "id",

"required": true,

"type": "integer",

"format": "int32"

}

],

"responses": {

"200": {

"description": "OK",

"schema": {

"$ref": "#/definitions/Task"

}

},

"401": {

"description": "Unauthorized"

},

"403": {

"description": "Forbidden"

},

"404": {

"description": "Not Found"

}

}

}

}

},

"definitions": {

"ModelAndView": {

"type": "object",

"properties": {

"empty": {

"type": "boolean"

},

"model": {

"type": "object"

},

"modelMap": {

"type": "object",

"additionalProperties": {

"type": "object"

}

},

"reference": {

"type": "boolean"

},

"status": {

"type": "string",

"enum": [

"ACCEPTED",

"ALREADY\_REPORTED",

"BAD\_GATEWAY",

"BAD\_REQUEST",

"BANDWIDTH\_LIMIT\_EXCEEDED",

"CHECKPOINT",

"CONFLICT",

"CONTINUE",

"CREATED",

"DESTINATION\_LOCKED",

"EXPECTATION\_FAILED",

"FAILED\_DEPENDENCY",

"FORBIDDEN",

"FOUND",

"GATEWAY\_TIMEOUT",

"GONE",

"HTTP\_VERSION\_NOT\_SUPPORTED",

"IM\_USED",

"INSUFFICIENT\_SPACE\_ON\_RESOURCE",

"INSUFFICIENT\_STORAGE",

"INTERNAL\_SERVER\_ERROR",

"I\_AM\_A\_TEAPOT",

"LENGTH\_REQUIRED",

"LOCKED",

"LOOP\_DETECTED",

"METHOD\_FAILURE",

"METHOD\_NOT\_ALLOWED",

"MOVED\_PERMANENTLY",

"MOVED\_TEMPORARILY",

"MULTIPLE\_CHOICES",

"MULTI\_STATUS",

"NETWORK\_AUTHENTICATION\_REQUIRED",

"NON\_AUTHORITATIVE\_INFORMATION",

"NOT\_ACCEPTABLE",

"NOT\_EXTENDED",

"NOT\_FOUND",

"NOT\_IMPLEMENTED",

"NOT\_MODIFIED",

"NO\_CONTENT",

"OK",

"PARTIAL\_CONTENT",

"PAYLOAD\_TOO\_LARGE",

"PAYMENT\_REQUIRED",

"PERMANENT\_REDIRECT",

"PRECONDITION\_FAILED",

"PRECONDITION\_REQUIRED",

"PROCESSING",

"PROXY\_AUTHENTICATION\_REQUIRED",

"REQUESTED\_RANGE\_NOT\_SATISFIABLE",

"REQUEST\_ENTITY\_TOO\_LARGE",

"REQUEST\_HEADER\_FIELDS\_TOO\_LARGE",

"REQUEST\_TIMEOUT",

"REQUEST\_URI\_TOO\_LONG",

"RESET\_CONTENT",

"SEE\_OTHER",

"SERVICE\_UNAVAILABLE",

"SWITCHING\_PROTOCOLS",

"TEMPORARY\_REDIRECT",

"TOO\_EARLY",

"TOO\_MANY\_REQUESTS",

"UNAUTHORIZED",

"UNAVAILABLE\_FOR\_LEGAL\_REASONS",

"UNPROCESSABLE\_ENTITY",

"UNSUPPORTED\_MEDIA\_TYPE",

"UPGRADE\_REQUIRED",

"URI\_TOO\_LONG",

"USE\_PROXY",

"VARIANT\_ALSO\_NEGOTIATES"

]

},

"view": {

"$ref": "#/definitions/View"

},

"viewName": {

"type": "string"

}

},

"title": "ModelAndView"

},

"Task": {

"type": "object",

"properties": {

"framework": {

"type": "string"

},

"id": {

"type": "integer",

"format": "int32"

},

"language": {

"type": "string"

},

"name": {

"type": "string"

}

},

"title": "Task"

},

"View": {

"type": "object",

"properties": {

"contentType": {

"type": "string"

}

},

"title": "View"

}

}

}

And now I have swagger ui to view all my documentation

Graphical user interface, text, application, email

Description automatically generated

And now lets try some of the api directly from swagger ui for example:

Get/task2/findAllServers:

Graphical user interface, application

Description automatically generated

In the screen shot you can see the response and status code of the directly generated by the swagger