**Node js Environment Variable settings:**

'use strict';

const config = {

development: {

admin: {

email: "babywarmersys@3v",

password: "baby^wrm"

},

storage: {

main: {

username: "root",

password: "ganesh",

database: "babywarmerdb",

host: "localhost",

// port:"3306",

dialect: "mysql",

dialectOptions: {

useUTC: false, // for reading from database

dateStrings: true,

typeCast: true

},

timezone: "+05:30",

collate: "utf8\_general\_ci",

pool: {

max: 100,

min: 0,

idle: 10000,

evict: 1000,

acquire: 30000,

handleDisconnects: true

},

retry: true

}

},

SYNC: false,

project: {

name: "babywarmer-dev",

secret: "BABYWARMER-secret-localserver"

},

server: {

connections: [{

port: 8080,

routes: {

cors: {

origin: ['\*'],

additionalHeaders: ['machineid', 'userid', 'secretuser', 'Authorization']

}

},

router: {

stripTrailingSlash: true

}

}],

registrations: [{

plugin: "hapi-auth-jwt2"

},

{

plugin: "./auth"

},

// {

// plugin: './developerauth'

// },

{

plugin: "./api",

options: {

routes: {

prefix: "/api/v1"

}

}

},

{

plugin: "inert"

},

{

plugin: "vision"

},

{

plugin: {

register: "hapi-swagger",

options: {

grouping: "tags",

securityDefinitions: {

Bearer: {

type: "apiKey",

name: "Authorization",

in: "header",

"x-keyPrefix": "Bearer "

}

}

// ,

// auth:'developer'

}

}

}

]

},

externals: {

smsApi: {},

usrmgt: {},

googlecaptchaapi: "https://www.google.com"

},

internals: {

systemkey: '15419O105143556356',

systemid: 12,

country: 'india',

countryid: 1,

countrycode: '+91',

isocode: 'IND',

routs: "all",

errorlogsourceserver: "stage",

tokenrandomcharlength: 3,

tokenencodetkey: "#D^FG3@4as$df^5#",

machinerandomkey: "%1349sbw^hm",

appsecretuser: "#M^LP3@4se$df^2#",

cloudsecretuser: "#Q^LP3@8le$df^8#",

randomkeylength: 3,

createormodifyby: "backendapis@babywarmer.com",

recaptchasecretkey: "6LcQla8ZAAAAACc0C3OO6qR-If0IBa5CLC4\_O3NB",

apifile: {

login: {

executionlog: true

}

}

},

schedular: {

aws: {

accessKeyId: "AKIA6G4DYAFLJVUVSOIU",

secretAccessKey: "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"

},

},

errMobileNo: ``,

aws: {

bucket\_name: 'app-iottechbucket',

bucket\_region: 'us-east-1',

signedurlexpires: 60 \* 60,

log\_bucket\_name: '',

log\_bucket\_region: ''

},

key: "123|a123123123123123@&"

},

test: {

admin: {

email: "babywarmersys@3v",

password: "baby^wrm"

},

storage: {

main: {

username: process.env.USERNAME,

password: process.env.PASSWORD,

database: process.env.DATABASE,

host: process.env.HOST,

timezone: process.env.TIMEZONE,

dialect: "",

collate: "utf8\_general\_ci"

}

},

project: {

name: "babywarmer-dev",

secret: process.env.APP\_SECRET || "BABYWARMER-secret-testserver"

},

server: {

connections: [{

port: process.env.SERVERPORT,

routes: {

cors: {

origin: ['\*'],

additionalHeaders: ['userid']

}

},

router: {

stripTrailingSlash: true

}

}],

registrations: [{

plugin: "hapi-auth-jwt2"

},

{

plugin: "./auth"

},

// {

// plugin: './developerauth'

// },

{

plugin: "./api",

options: {

routes: {

prefix: "/api/v1"

}

}

},

{

plugin: "inert"

},

{

plugin: "vision"

},

{

plugin: {

register: "hapi-swagger",

options: {

grouping: "tags",

securityDefinitions: {

Bearer: {

type: "apiKey",

name: "Authorization",

in: "header",

"x-keyPrefix": "Bearer "

}

}

// schemes: ['https'],

// auth:'developer'

}

}

}

]

},

externals: {

smsApi: {},

usrmgt: {},

googlecaptchaapi: process.env.GOOGLECAPTCHAAPI

},

internals: {

systemkey: process.env.SYSTEMKEY,

systemid: process.env.SYSTEMID,

country: process.env.COUNTRY,

countryid: process.env.COUNTRYID,

countrycode: process.env.COUNTRYCODE,

isocode: process.env.ISOCODE,

routs: process.env.ROUTS,

errorlogsourceserver: process.env.ERRORLOGSOURCESERVER,

tokenrandomcharlength: process.env.TOKENRANDOMCHARLENGTH,

tokenencodetkey: process.env.TOKENENCODETKEY,

recaptchasecretkey: process.env.RECAPTCHASECRETKEY,

apifile: process.env.APIFILE

},

schedular: {

aws: {

accessKeyId: "AKIA6G4DYAFLJVUVSOIU",

secretAccessKey: "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"

},

},

errMobileNo: ``,

aws: {

bucket\_name: 'app-iottechbucket',

bucket\_region: 'us-east-1',

signedurlexpires: 60 \* 60,

log\_bucket\_name: '',

log\_bucket\_region: ''

},

key: "123|a123123123123123@&"

// schedular: {

// aws: {

// accessKeyId: process.env.ACCESSKEYID,

// secretAccessKey: process.env.SECRETACCESSKEY

// }

// },

// errMobileNo: '',

// aws: {

// bucket\_name: process.env.BUCKETNAME,

// bucket\_region: process.env.BUCKETREGION,

// signedurlexpires: process.env.SIGNEDURLEXPIRES,

// log\_bucket\_name: process.env.LOGBUCKETNAME,

// log\_bucket\_region: process.env.LOGBUCKETREGION

// },

// key: process.env.KEY

},

production: {

admin: {

email: "babywarmersys@3v",

password: "baby^wrm"

},

storage: {

main: {

username: process.env.USERNAME,

password: process.env.PASSWORD,

database: process.env.DATABASE,

host: process.env.HOST,

dialect: process.env.DIALECT,

dialectOptions: {

useUTC: false, // for reading from database

dateStrings: true,

typeCast: true

},

timezone: "+05:30",

collate: "utf8\_general\_ci"

}

},

project: {

name: "babywarmer-dev",

secret: process.env.APP\_SECRET || "BABYWARMER-secret-prodserver"

},

server: {

connections: [{

port: process.env.SERVERPORT,

routes: {

cors: {

origin: ['\*'],

additionalHeaders: ['machineid', 'userid', 'secretuser', 'Authorization']

}

},

router: {

stripTrailingSlash: true

}

}],

registrations: [{

plugin: "hapi-auth-jwt2"

},

{

plugin: "./auth"

},

// {

// plugin: './developerauth'

// },

{

plugin: "./api",

options: {

routes: {

prefix: "/api/v1"

}

}

},

{

plugin: "inert"

},

{

plugin: "vision"

},

{

plugin: {

register: "hapi-swagger",

options: {

grouping: "tags",

securityDefinitions: {

Bearer: {

type: "apiKey",

name: "Authorization",

in: "header",

"x-keyPrefix": "Bearer "

}

}

// schemes: ['https'],

// auth:'developer'

}

}

}

]

},

externals: {

smsApi: {},

usrmgt: {},

googlecaptchaapi: process.env.GOOGLECAPTCHAAPI

},

internals: {

systemkey: process.env.SYSTEMKEY,

systemid: process.env.SYSTEMID,

country: process.env.COUNTRY,

countryid: process.env.COUNTRYID,

countrycode: process.env.COUNTRYCODE,

isocode: process.env.ISOCODE,

routs: process.env.ROUTS,

errorlogsourceserver: process.env.ERRORLOGSOURCESERVER,

tokenrandomcharlength: process.env.TOKENRANDOMCHARLENGTH,

tokenencodetkey: process.env.TOKENENCODETKEY,

appsecretuser: process.env.APPSECRETUSER,

cloudsecretuser: process.env.CLOUDSECRETUSER,

createormodifyby: process.env.CREATEORMODIFYBY,

otplength: process.env.OTPLENGTH,

recaptchasecretkey: process.env.RECAPTCHASECRETKEY,

apifile: process.env.APIFILE

},

schedular: {

aws: {

accessKeyId: "AKIA6G4DYAFLJVUVSOIU",

secretAccessKey: "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"

},

},

errMobileNo: ``,

aws: {

bucket\_name: 'app-iottechbucket',

bucket\_region: 'us-east-1',

signedurlexpires: 60 \* 60,

log\_bucket\_name: '',

log\_bucket\_region: ''

},

key: "123|a123123123123123@&"

// schedular: {

// aws: {

// accessKeyId: process.env.ACCESSKEYID ,

// secretAccessKey: process.env.SECRETACCESSKEY

// }

// },

// errMobileNo: '',

// aws: {

// bucket\_name: process.env.BUCKETNAME,

// bucket\_region: process.env.BUCKETREGION,

// signedurlexpires: process.env.SIGNEDURLEXPIRES,

// log\_bucket\_name: process.env.LOGBUCKETNAME,

// log\_bucket\_region: process.env.LOGBUCKETREGION

// },

// key: process.env.KEY

}

};

const env = process.env.NODE\_ENV || 'development';

config.env = env;

module.exports = config[env];