

# Primeware.Trex.Calc.Api

v1

OAS3

</swagger/v1/swagger.json>

## Calculate

**GET**[/api/Calculate/models](#) Get all of heatexchanger model names

### Parameters

[Try it out](#)

Name	Description
------	-------------

License_Key	You should have License key to use this endpoint
-------------	--

**String(\$uuid)**  
*(header)*

License\_Key - You should have License key ·

### Responses

Code	Description	Links
------	-------------	-------

200	Success	No links
-----	---------	----------

Media type

**text/plain**



Controls Accept header.

[Example Value](#) | [Schema](#)

```
{  
  "version": "string",  
  "statusCode": 0,  
  "message": "string",  
  "isError": true  
}
```

**POST**[/api/Calculate](#) Calculate for two season in the same time

Name	Description
------	-------------

## License\_Key

String(\$uuid)  
(header)

You should have License key to use this endpoint

License\_Key - You should have License key

## Request body

application/json

## Example Value | Schema

```
{  
    "summer": {  
        "airFlowEx": 0,  
        "tempEx": 0,  
        "rhEx": 0,  
        "airFlowSup": 0,  
        "tempSup": 0,  
        "rhSup": 0  
    },  
    "winter": {  
        "airFlowEx": 0,  
        "tempEx": 0,  
        "rhEx": 0,  
        "airFlowSup": 0,  
        "tempSup": 0,  
        "rhSup": 0  
    },  
    "heatExchanger": {  
        "modelCode": "string",  
        "profitType": 0,  
        "coverType": 0,  
        "depth": 0,  
        "altitude": 0,  
        "isDensityConst": true  
    }  
}
```

## Responses

Code	Description	Links
------	-------------	-------

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accept header.

[Example Value](#) | [Schema](#)

```
{
  "version": "string",
  "statusCode": 0,
  "message": "string",
  "isError": true,
  "trexOutputs": {
    "heatExchangerOutput": {
      "a": 0,
      "b": 0,
      "c": 0,
      "length": 0,
      "weight": 0,
      "pieceCount": 0,
      "price": 0,
      "orderCode": "string"
    },
    "summerAirOutput": {
      "supplyVelocityIn": 0,
      "supplyVelocityOut": 0,
      "exhaustVelocityIn": 0,
      "exhaustVelocityOut": 0,
      "supplyStandartVelocity": 0,
      "exhaustStandartVelocity": 0,
      "supplyAirDensityIn": 0,
      "supplyAirDensityOut": 0,
      "exhaustAirDensityIn": 0,
      "exhaustAirDensityOut": 0,
      "supplyEffectiveAirFlowIn": 0,
      "exhaustEffectiveAirFlowOut": 0
    }
  }
}
```

**POST** /api/Calculate/summer Calculate for summer season

Parameters		<a href="#">Try it out</a>
Name	Description	
License_Key String(\$uuid) (header)	You should have License key to use this endpoint	
License_Key - You should have License key		
Request body	<a href="#">application/json</a>	

[Example Value](#) | [Schema](#)

```
{
  "summer": {
```

```

    "airFlowEx": 0,
    "tempEx": 0,
    "rhEx": 0,
    "airFlowSup": 0,
    "tempSup": 0,
    "rhSup": 0
},
"heatExchanger": {
    "modelCode": "string",
    "profitType": 0,
    "coverType": 0,
    "depth": 0,
    "altitude": 0,
    "isDensityConst": true
}
}

```

## Responses

Code	Description	Links
200	<p>Success</p> <p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">text/plain</div> <p>Controls Accept header.</p> <p><a href="#">Example Value</a>   <a href="#">Schema</a></p> <pre>{     "version": "string",     "statusCode": 0,     "message": "string",     "isError": true,     "trexOutputs": {         "heatExchangerOutput": {             "a": 0,             "b": 0,             "c": 0,             "length": 0,             "weight": 0,             "pieceCount": 0,             "price": 0,             "orderCode": "string"         },         "summerAirOutput": {             "supplyVelocityIn": 0,             "supplyVelocityOut": 0,             "exhaustVelocityIn": 0,             "exhaustVelocityOut": 0,             "supplyStandartVelocity": 0,             "exhaustStandartVelocity": 0,             "supplyAirDensityIn": 0,             "supplyAirDensityOut": 0,             "exhaustAirDensityIn": 0,             "exhaustAirDensityOut": 0,             "supplyEffectiveAirFlowIn": 0,             "exhaustEffectiveAirFlowOut": 0         }     } }</pre>	<i>No links</i>

POST

/api/Calculate/winter Calculate for winter season

Name	Description
------	-------------

## License\_Key

String(\$uuid)  
(header)

You should have License key to use this endpoint

License\_Key - You should have License key

## Request body

application/json

Example Value | Schema

```
{  
  "winter": {  
    "airFlowEx": 0,  
    "tempEx": 0,  
    "rhEx": 0,  
    "airFlowSup": 0,  
    "tempSup": 0,  
    "rhSup": 0  
  },  
  "heatExchanger": {  
    "modelCode": "string",  
    "profitType": 0,  
    "coverType": 0,  
    "depth": 0,  
    "altitude": 0,  
    "isDensityConst": true  
  }  
}
```

## Responses

Code	Description	Links
------	-------------	-------

Code	Description	Links
200	<p>Success</p> <p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">text/plain</div> <p>Controls Accept header.</p> <p><a href="#">Example Value</a>   <a href="#">Schema</a></p> <pre>{   "version": "string",   "statusCode": 0,   "message": "string",   "isError": true,   "trexOutputs": {     "heatExchangerOutput": {       "a": 0,       "b": 0,       "c": 0,       "length": 0,       "weight": 0,       "pieceCount": 0,       "price": 0,       "orderCode": "string"     },     "summerAirOutput": {       "supplyVelocityIn": 0,       "supplyVelocityOut": 0,       "exhaustVelocityIn": 0,       "exhaustVelocityOut": 0,       "supplyStandartVelocity": 0,       "exhaustStandartVelocity": 0,       "supplyAirDensityIn": 0,       "supplyAirDensityOut": 0,       "exhaustAirDensityIn": 0,       "exhaustAirDensityOut": 0,       "supplyEffectiveAirFlowIn": 0,       "exhaustEffectiveAirFlowOut": 0     }   } }</pre>	No links

## Schemas



```
ApiResponse ▾ {
  version           string
  nullable: true
  statusCode       integer($int32)
  message          string
  nullable: true
  isError          boolean
  nullable: true
  responseException > {...}
  nullable: true
  result           > {...}
  nullable: true
}
```

```
TrexInputAirData ▾ {  
    airFlowEx           number($float)  
                        Air flow of exhaust side. Range: (200,100000) m^3/h  
    tempEx             number($float)  
                        Temperature of exhaust side. Range:[-50,150] C  
    rhEx               number($float)  
                        Relative humidity of exhaust side. Range:[0,100] %  
    airFlowSup          number($float)  
                        Air flow of supply side. Range: (200,100000) m^3/h  
    tempSup             number($float)  
                        Temperature of supply side. Range:[-50,150] C  
    rhSup               number($float)  
                        Relative humidity of supply side. Range:[0,100] %  
}  
}
```

**HEProfitType** integer(\$int32)

Profit type of heat exchanger. We have two profit type. S profit type S=0, G profit type G=1

Enum:

▼ [ 0, 1 ]

**HECoverType** integer(\$int32)

Cover type of heat exchanger. We have two cover type. Standart cover type=0, Right cover type=1

Enum:

▼ [ 0, 1 ]

```

TrexInputHeatExchanger ▼ {
    modelCode          string
    nullable: true

        Heat exchanger model code which embedded on system. You can get those codes use
        this '/api/Calculate/models'

    profitType        HEProfitType integer($int32)

        Profit type of heat exchanger. We have two profit type. S profit type S=0, G
        profit type G=1

        Enum:
            ▼ [ 0, 1 ]
    coverType          HECoverType integer($int32)

        Cover type of heat exchanger. We have two cover type. Standard cover type=0,
        Right cover type=1

        Enum:
            ▼ [ 0, 1 ]
    depth              number($float)

        Depth of heat exchanger. Range: (200,4500) mm

    altitude           number($float)

        Altitude of heat exchanger. Range: (-10000,10000) m

    isDensityConst     boolean

        if you want density to be const you must set true value to IsDensityConst
        property. Default value is true

}

```

```

TrexInputsForTwoSeason ▼ {
    summer             TrexInputAirData ▼ {
        airFlowEx         number($float)

            Air flow of exhaust side. Range: (200,100000) m^3/h

        tempEx            number($float)

            Temperature of exhaust side. Range:[-50,150] C

        rhEx              number($float)

            Relative humidity of exhaust side. Range:[0,100] %

        airFlowSup         number($float)

            Air flow of supply side. Range: (200,100000) m^3/h

        tempSup           number($float)

            Temperature of supply side. Range:[-50,150] C

        rhSup              number($float)

            Relative humidity of supply side. Range:[0,100] %

    }
    winter             TrexInputAirData ▼ {
        airFlowEx         number($float)

            Air flow of exhaust side. Range: (200,100000) m^3/h

        tempEx            number($float)

            Temperature of exhaust side. Range:[-50,150] C

        rhEx              number($float)

```

```

    rhEx          number($float)           Relative humidity of exhaust side. Range:[0,100] %

    airFlowSup    number($float)           Air flow of supply side. Range: (200,100000) m^3/h

    tempSup      number($float)           Temperature of supply side. Range:[-50,150] C

    rhSup         number($float)           Relative humidity of supply side. Range:[0,100] %

}

heatExchanger
TrexInputHeatExchanger ▼ {
    modelCode     string                nullable: true
                                Heat exchanger model code which embedded on system. You
                                can get those codes use this '/api/Calculate/models'

    profitType   HEProfitType integer($int32)
                                Profit type of heat exchanger. We have two profit type.
                                S profit type S=0, G profit type G=1

    coverType    HECoverType integer($int32)
                                Cover type of heat exchanger. We have two cover type.
                                Standart cover type=0, Right cover type=1

    depth        number($float)
                                Depth of heat exchanger. Range: (200,4500) mm

    altitude     number($float)
                                Altitude of heat exchanger. Range: (-10000,10000) m

    isDensityConst boolean
                                if you want density to be const you must set true value
                                to IsDensityConst property. Default value is true

}
}

```

```
TrexOutputHeatExchanger ▼ {
    a
        number($float)
            Edge of Heat exchanger Unit:[mm]
    b
        number($float)
            Edge of Heat exchanger. Unit:[mm]
    c
        number($float)
            Diagonal of Heat exchanger. Unit:[mm]
    length
        number($float)
            Length of Heat exchanger. Unit:[mm]
    weight
        number($float)
            Weight of Heat exchanger. Unit:[kg]
    pieceCount
        integer($int32)
            Piece count of Heat exchanger
    price
        number($float)
            Price of Heat exchanger. Unit:[USD]
    orderCode
        string
            nullable: true
            Order code of Heat exchanger
}
```

```

TrexOutputAirData ▼ {
    supplyVelocityIn          number($float)
    supplyVelocityOut         number($float)
    exhaustVelocityIn         number($float)
    exhaustVelocityOut        number($float)
    supplyStandartVelocity   number($float)
    exhaustStandartVelocity  number($float)
    supplyAirDensityIn       number($float)
    supplyAirDensityOut      number($float)
    exhaustAirDensityIn      number($float)
    exhaustAirDensityOut     number($float)
    supplyEffectiveAirFlowIn number($float)
    supplyEffectiveAirFlowOut number($float)
    exhaustEffectiveAirFlowIn number($float)
    exhaustEffectiveAirFlowOut number($float)
    supplyStandartAirFlow    number($float)
    exhaustStandartAirFlow   number($float)
    supplyMassFlow            number($float)
    exhaustMassFlow           number($float)
    supplyTempIn              number($float)
    supplyTempOut             number($float)
    exhaustTempIn             number($float)
    exhaustTempOut            number($float)
    supplyRhIn                number($float)
    supplyRhOut               number($float)
    exhaustRhIn               number($float)
    exhaustRhOut              number($float)
    supplyWetBubIn            number($float)
    supplyWetBubOut           number($float)
    exhaustWetBubIn           number($float)
    exhaustWetBubOut          number($float)
    supplyWetEfficiency      number($float)
    exhaustWetEfficiency     number($float)
    supplyDryEfficiency       number($float)
    exhaustDryEfficiency      number($float)
    supplyWetEffectivness    number($float)
    exhaustWetEffectivness   number($float)
    supplyDryEffectivness    number($float)
    exhaustDryEffectivness   number($float)
    dewpoint                 number($float)
    hrsCapacity               number($float)
    supplyPressureDrop        number($float)
    exhaustPressureDrop       number($float)
    supplyStandartPressureDrop number($float)
    exhaustStandartPressureDrop number($float)
    condensationAmount       number($float)
    energyEfficiency          number($float)
    heatRecoveryClass         string
                                nullable: true
    thermalDryEfficiency     number($float)
    supplyEnthalpyIn          number($float)
    supplyEnthalpyOut         number($float)
    exhaustEnthalpyIn         number($float)
    exhaustEnthalpyOut        number($float)
    supplyHumRatioIn          number($float)
    supplyHumRatioOut         number($float)
    exhaustHumRatioIn         number($float)
    exhaustHumRatioOut        number($float)
    atm                       number($float)
    altitude                  number($float)
    warningText                string
                                nullable: true
}

```

TrexOutputs ▼ {  
 heatExchangerOutput

TrexOutputHeatExchanger ▼ {  
 a number(\$float)

Edge of Heat exchanger Unit:[mm]

b number(\$float)

```

    c                                Edge of Heat exchanger. Unit:[mm]
    number($float)

    length                           Diagonal of Heat exchanger. Unit:[mm]
    number($float)

    weight                            Length of Heat exchanger. Unit:[mm]
    number($float)

    weight                            Weight of Heat exchanger. Unit:[kg]
    number($float)

    pieceCount                        pieceCount of Heat exchanger
    integer($int32)

    price                             Price of Heat exchanger. Unit:[USD]
    number($float)

    orderCode                          Order code of Heat exchanger
    string
    nullable: true

}
summerAirOutput

```

### TrexOutputAirData ▼ {

supplyVelocityIn	number(\$float)
supplyVelocityOut	number(\$float)
exhaustVelocityIn	number(\$float)
exhaustVelocityOut	number(\$float)
supplyStandartVelocity	number(\$float)
exhaustStandartVelocity	number(\$float)
supplyAirDensityIn	number(\$float)
supplyAirDensityOut	number(\$float)
exhaustAirDensityIn	number(\$float)
exhaustAirDensityOut	number(\$float)
supplyEffectiveAirFlowIn	number(\$float)
supplyEffectiveAirFlowOut	number(\$float)
exhaustEffectiveAirFlowIn	number(\$float)
exhaustEffectiveAirFlowOut	number(\$float)
supplyStandartAirFlow	number(\$float)
exhaustStandartAirFlow	number(\$float)
supplyMassFlow	number(\$float)
exhaustMassFlow	number(\$float)
supplyTempIn	number(\$float)
supplyTempOut	number(\$float)
exhaustTempIn	number(\$float)
exhaustTempOut	number(\$float)
supplyRhIn	number(\$float)
supplyRhOut	number(\$float)
exhaustRhIn	number(\$float)
exhaustRhOut	number(\$float)
supplyWetBubIn	number(\$float)
supplyWetBubOut	number(\$float)
exhaustWetBubIn	number(\$float)
exhaustWetBubOut	number(\$float)
supplyWetEfficiency	number(\$float)
exhaustWetEfficiency	number(\$float)
supplyDryEfficiency	number(\$float)
exhaustDryEfficiency	number(\$float)
supplyNetEffectivness	number(\$float)
exhaustNetEffectivness	number(\$float)
supplyDryEffectivness	number(\$float)
exhaustDryEffectivness	number(\$float)
dewpoint	number(\$float)
hrsCapacity	number(\$float)
supplyPressureDrop	number(\$float)
exhaustPressureDrop	number(\$float)
supplyStandartPressureDrop	number(\$float)
exhaustStandartPressureDrop	number(\$float)
condensationAmount	number(\$float)
energyEfficiency	number(\$float)
heatRecoveryClass	string     nullable: true

```

    thermalDryEfficiency      number($float)
    supplyEnthalpyIn          number($float)
    supplyEnthalpyOut         number($float)
    exhaustEnthalpyIn         number($float)
    exhaustEnthalpyOut        number($float)
    supplyHumRatioIn          number($float)
    supplyHumRatioOut         number($float)
    exhaustHumRatioIn         number($float)
    exhaustHumRatioOut        number($float)
    atm                         number($float)
    altitude                   number($float)
    warningText                string
                                nullable: true
}

winterAirOutput
TrexOutputAirData ▼ {
    supplyVelocityIn          number($float)
    supplyVelocityOut         number($float)
    exhaustVelocityIn         number($float)
    exhaustVelocityOut        number($float)
    supplyStandartVelocity    number($float)
    exhaustStandartVelocity   number($float)
    supplyAirDensityIn        number($float)
    supplyAirDensityOut       number($float)
    exhaustAirDensityIn       number($float)
    exhaustAirDensityOut      number($float)
    supplyEffectiveAirFlowIn  number($float)
    supplyEffectiveAirFlowOut number($float)
    exhaustEffectiveAirFlowIn number($float)
    exhaustEffectiveAirFlowOut number($float)
    supplyStandartAirFlow     number($float)
    exhaustStandartAirFlow    number($float)
    supplyMassFlow             number($float)
    exhaustMassFlow            number($float)
    supplyTempIn               number($float)
    supplyTempOut              number($float)
    exhaustTempIn              number($float)
    exhaustTempOut             number($float)
    supplyRhIn                number($float)
    supplyRhOut               number($float)
    exhaustRhIn               number($float)
    exhaustRhOut              number($float)
    supplyWetBubIn            number($float)
    supplyWetBubOut           number($float)
    exhaustWetBubIn           number($float)
    exhaustWetBubOut          number($float)
    supplyWetEfficiency       number($float)
    exhaustWetEfficiency      number($float)
    supplyDryEfficiency       number($float)
    exhaustDryEfficiency      number($float)
    supplyWetEffectivness     number($float)
    exhaustWetEffectivness    number($float)
    supplyDryEffectivness     number($float)
    exhaustDryEffectivness    number($float)
    dewpoint                  number($float)
    hrsCapacity               number($float)
    supplyPressureDrop         number($float)
    exhaustPressureDrop        number($float)
    supplyStandartPressureDrop number($float)
    exhaustStandartPressureDrop number($float)
    condensationAmount        number($float)
    energyEfficiency          number($float)
    heatRecoveryClass          string
                                nullable: true
    thermalDryEfficiency      number($float)
    supplyEnthalpyIn          number($float)
    supplyEnthalpyOut         number($float)
    exhaustEnthalpyIn         number($float)
    exhaustEnthalpyOut        number($float)
    supplyHumRatioIn          number($float)
    supplyHumRatioOut         number($float)
    exhaustHumRatioIn         number($float)
    exhaustHumRatioOut        number($float)
    atm                         number($float)
    altitude                   number($float)
    warningText                string
                                nullable: true
}

```

```
}
```

```
ApiCalculateResult ▼ {  
    version           string  
                      nullable: true  
    statusCode       integer($int32)  
    message          string  
                      nullable: true  
    isError          boolean  
                      nullable: true  
    responseException  
        ▼ {  
            }  
            nullable: true  
    result           ▼ {  
        }  
        nullable: true  
    trexOutputs      TrexOutputs > {...}  
}
```

```

TrexInputSummer ▼ {
    summer
        TrexInputAirData ▼ {
            airFlowEx           number($float)
                                Air flow of exhaust side. Range: (200,100000) m^3/h

            tempEx             number($float)
                                Temperature of exhaust side. Range:[-50,150] C

            rhEx               number($float)
                                Relative humidity of exhaust side. Range:[0,100] %

            airFlowSup          number($float)
                                Air flow of supply side. Range: (200,100000) m^3/h

            tempSup             number($float)
                                Temperature of supply side. Range:[-50,150] C

            rhSup               number($float)
                                Relative humidity of supply side. Range:[0,100] %

        }
    heatExchanger
        TrexInputHeatExchanger ▼ {
            modelCode           string
                                nullable: true
                                Heat exchanger model code which embedded on system. You can get those codes use this '/api/Calculate/models'

            profitType          HEProfitType integer($int32)
                                Profit type of heat exchanger. We have two profit type. S profit type S=0, G profit type G=1

            coverType            Enum:
                                > Array [ 2 ]
                                HECoverType integer($int32)
                                Cover type of heat exchanger. We have two cover type. Standart cover type=0, Right cover type=1

            depth                Enum:
                                > Array [ 2 ]
                                number($float)
                                Depth of heat exchanger. Range: (200,4500) mm

            altitude              number($float)
                                Altitude of heat exchanger. Range: (-10000,10000) m

            isDensityConst         boolean
                                if you want density to be const you must set true value to IsDensityConst property. Default value is true

        }
    }
}

```

```
TrexInputWinter ▼ {  
    winter  
        TrexInputAirData ▼ {  
            airFlowEx          number($float)  
                                Air flow of exhaust side. Range: (200,100000) m^3/h  
            tempEx           number($float)  
                                Temperature of exhaust side. Range:[-50,150] C  
            rhEx             number($float)  
                                Relative humidity of exhaust side. Range:[0,100] %  
            airFlowSup       number($float)  
                                Air flow of supply side. Range: (200,100000) m^3/h  
            tempSup          number($float)  
                                Temperature of supply side. Range:[-50,150] C  
            rhSup            number($float)  
                                Relative humidity of supply side. Range:[0,100] %  
        }  
    heatExchanger  
}  
    TrexInputHeatExchanger > {...}  
}
```