Infosys
Responsible AI Office

Infosys Responsible AI Toolkit – Privacy tenet

API usage Instructions

Contents

Introduction

AI systems often rely on vast amounts of personal data. To protect individual privacy, techniques like anonymization, masking, hashing, encryption, and differential privacy are employed to obscure or transform data. This ensures that sensitive information remains confidential while still allowing AI solutions provide valuable insights.

Once API swagger page is populated as per instructions given in the github repository Readme file, click on 'try it out' to use required endpoints. Details of endpoints associated with Privacy tenet are outlined below.

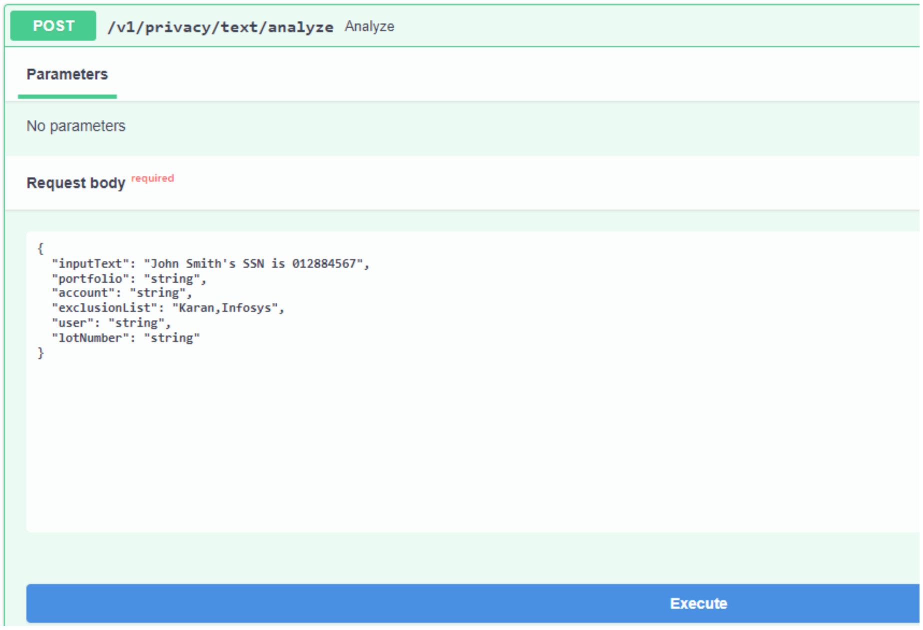
Analyze

Endpoint – /rai/v1/privacy/text/analyze
Using this API, we can check if the input text contains any PII entities or not.

Input :

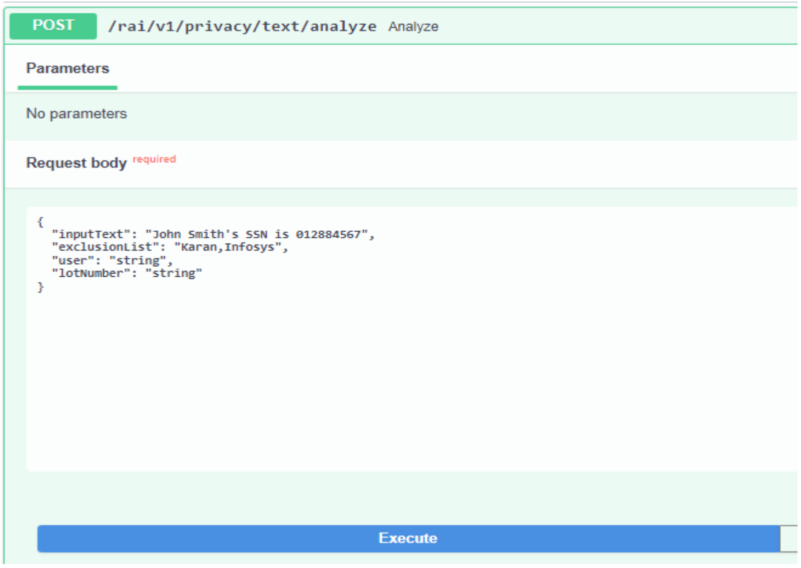
Replace the input text with the prompt you want to check for PII entities. In exclusion list mention the PII entities which don't need to be blocked. The fields portfolio and account are optional (we can remove these from the Json if these are not needed), and we can create these from the admin portal. The fields user and lotNumber will be allocated at user login if they are using the application.

With all fields -



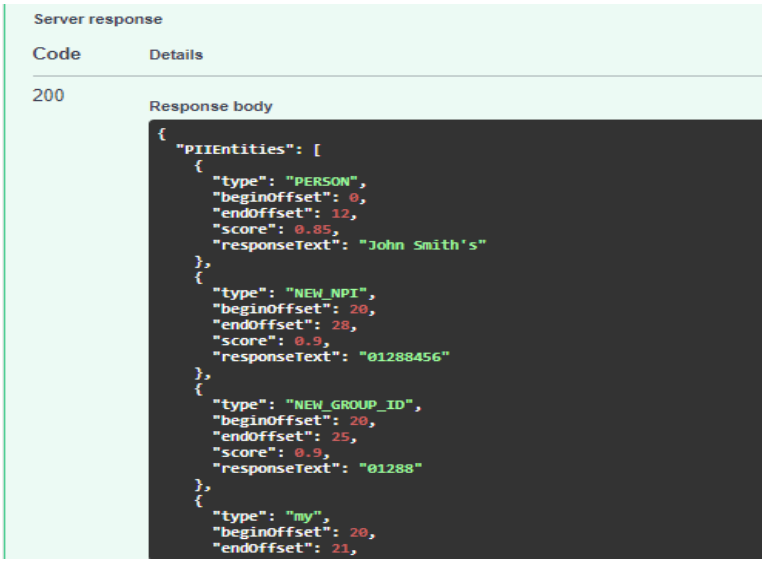
```
{
  "inputText": "John Smith's SSN is 012884567",
  "portfolio": "string",
  "account": "string",
  "exclusionList": "Karan,Infosys",
  "user": "string",
  "lotNumber": "string"
}
```

Without optional fields -



```
{
  "inputText": "John Smith's SSN is 012884567",
  "exclusionList": "Karan,Infosys",
  "user": "string",
  "lotNumber": "string"
}
```

Response :

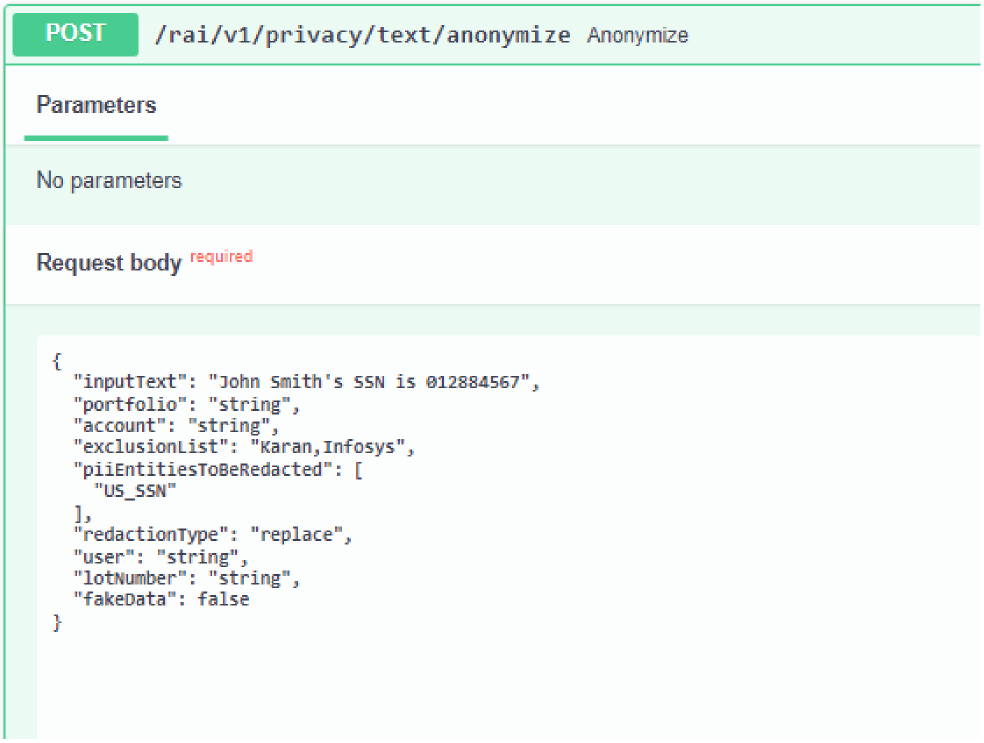


```
{
  "PIIEntities": [
    {
      "type": "PERSON",
      "beginOffset": 0,
      "endOffset": 12,
      "score": 0.85,
      "responseText": "John Smith's"
    },
    {
      "type": "NEW_NPI",
      "beginOffset": 20,
      "endOffset": 28,
      "score": 0.9,
      "responseText": "01288456"
    },
    {
      "type": "NEW_GROUP_ID",
      "beginOffset": 20,
      "endOffset": 25,
      "score": 0.9,
      "responseText": "01288"
    },
    {
      "type": "my",
      "beginOffset": 20,
      "endOffset": 21,
```

Anonymize

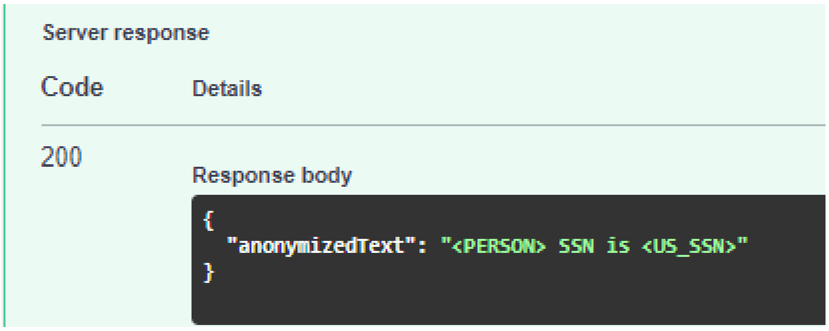
Endpoint – /rai/v1/privacy/text/anonymize
Using this API, we can anonymize all PII entities in the input text.

Input : Replace the input text with prompt to be checked, give the PII entities to be redacted in the list. Give fakeData as True if you want to mask the PII detected with fake data. The fields portfolio and account are optional (we can remove these from the Json if these are not needed), and we can create these from the admin portal. The fields user and lotNumber will be allocated at user login if they are using the application. We can give redaction type for the type of anonymization we want and can give PII entities in the list which we want to be redacted. In exclusion list mention the PII values which don't need to be anonymized and identified.



```
{
  "inputText": "John Smith's SSN is 012884567",
  "portfolio": "string",
  "account": "string",
  "exclusionList": "Karan,Infosys",
  "piiEntitiesToBeRedacted": [
    "US_SSN"
  ],
  "redactionType": "replace",
  "user": "string",
  "lotNumber": "string",
  "fakeData": false
}
```

Response :



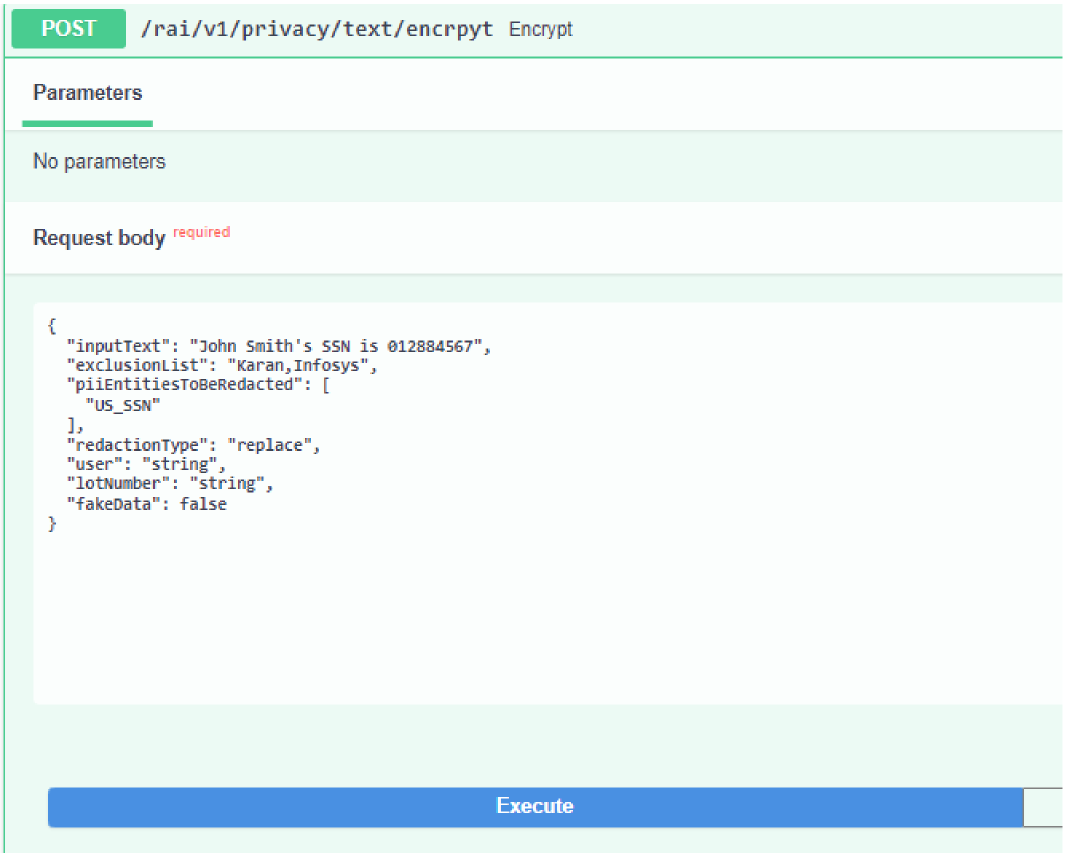
```
{
  "anonymizedText": "<PERSON> SSN is <US_SSN>"
}
```

Encrpyt

Endpoint – /rai/v1/privacy/text/encrpyt
Using this API, we can encrypt all PII entities in the input text.

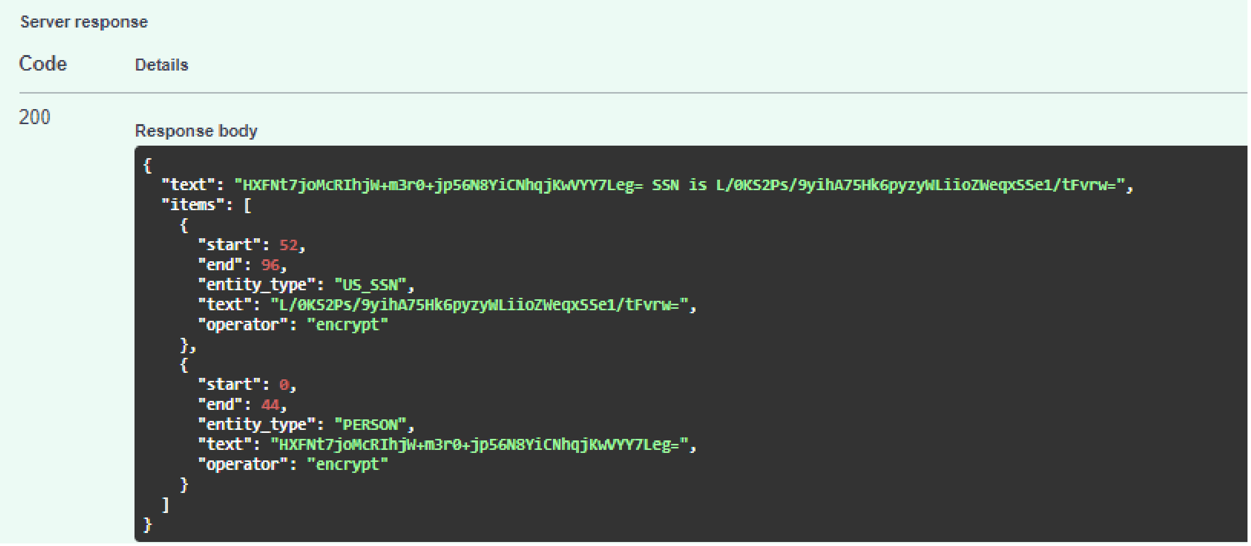
Input :
Replace the input text with prompt to be checked. Give fakeData as True if you want to mask the PII detected with fake data. The fields portfolio and account are optional (we can remove these from the Json if these are not needed), and we can create these from the admin portal. The fields user and lotNumber will be allocated at user login if they are using the application. We can give

redaction type for the type of anonymization we want and can give PII entities in the list which we want to be identified and encrypted.



```
{
  "inputText": "John Smith's SSN is 012884567",
  "exclusionList": "Karan,Infosys",
  "piiEntitiesToBeRedacted": [
    "US_SSN"
  ],
  "redactionType": "replace",
  "user": "string",
  "lotNumber": "string",
  "fakeData": false
}
```

Response :



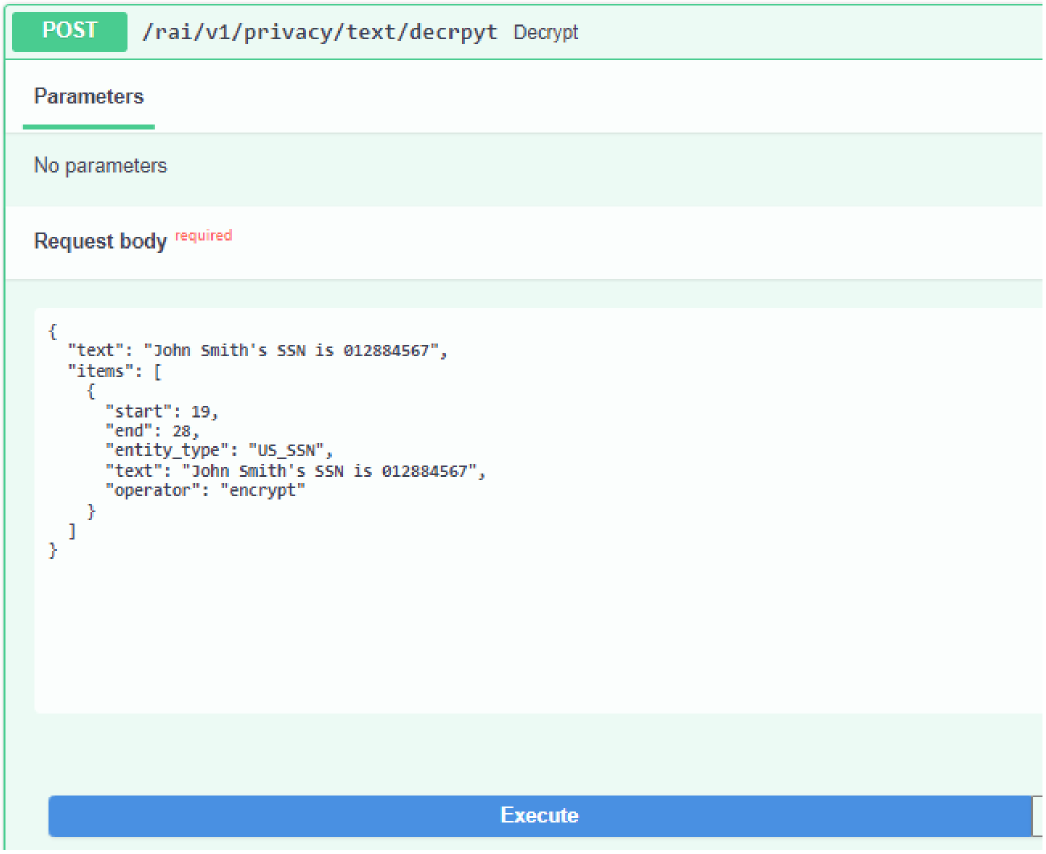
```
{
  "text": "HXFNt7joMcRIhjW+m3r0+jp56N8YiCNhqjKwVYY7Leg= SSN is L/0KS2Ps/9yihA75Hk6pyzyWLiioZWeqxSSe1/tFvrw=",
  "items": [
    {
      "start": 52,
      "end": 96,
      "entity_type": "US_SSN",
      "text": "L/0KS2Ps/9yihA75Hk6pyzyWLiioZWeqxSSe1/tFvrw=",
      "operator": "encrypt"
    },
    {
      "start": 0,
      "end": 44,
      "entity_type": "PERSON",
      "text": "HXFNt7joMcRIhjW+m3r0+jp56N8YiCNhqjKwVYY7Leg=",
      "operator": "encrypt"
    }
  ]
}
```

Decrypt

Endpoint – /rai/v1/privacy/text/decrpyt

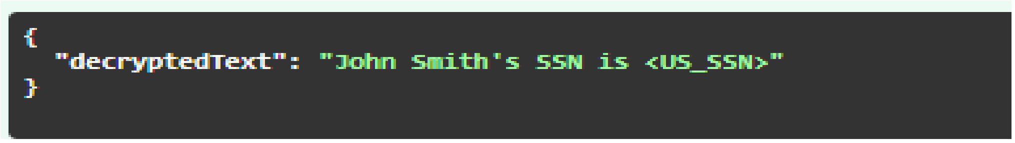
Using this API, we can encrypt PII entities of a specific type at given location in the text.

Input : Replace the input text with prompt to be checked, give the PII entities to be encrypted in the list.



```
{
  "text": "John Smith's SSN is 012884567",
  "items": [
    {
      "start": 19,
      "end": 28,
      "entity_type": "US_SSN",
      "text": "John Smith's SSN is 012884567",
      "operator": "encrypt"
    }
  ]
}
```

Response :



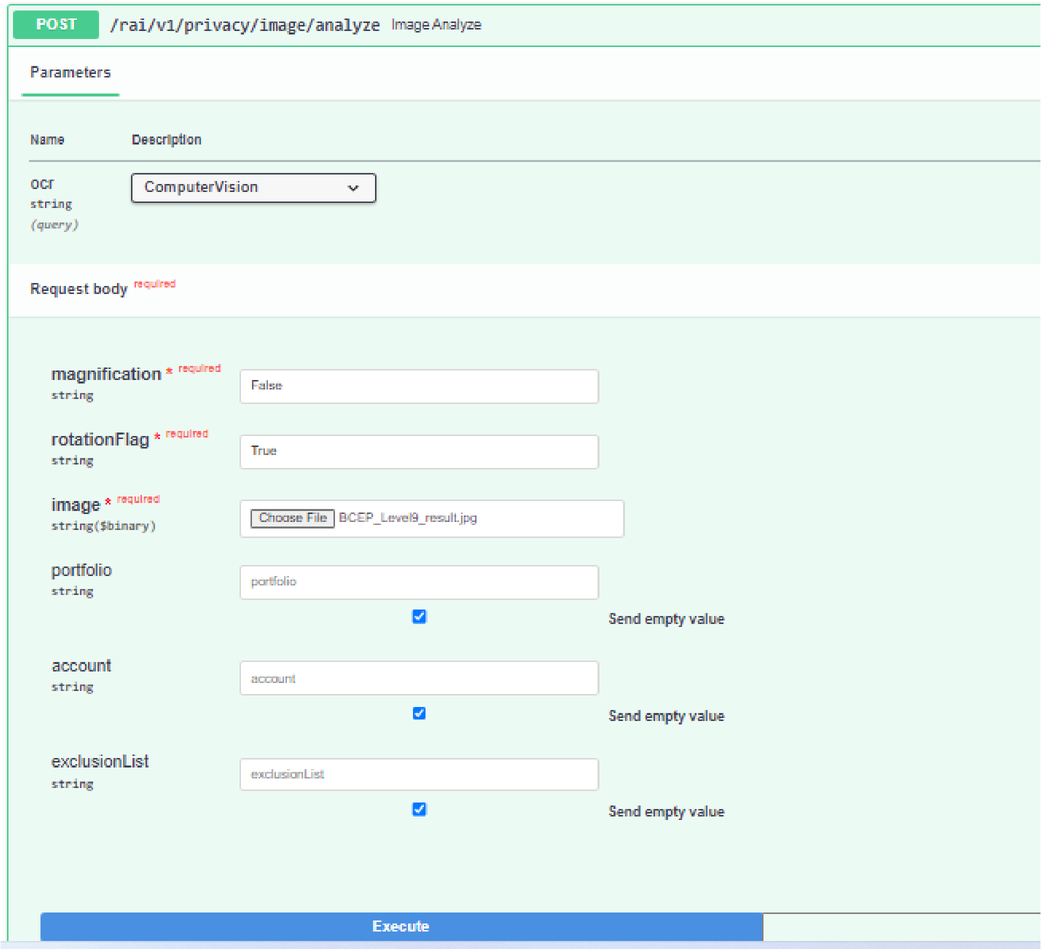
```
{
  "decryptedText": "John Smith's SSN is <US_SSN>"
}
```

Image Analyze

Endpoint – /rai/v1/privacy/image/analyze
Using this API, we can analyze the uploaded image for any PII entities.

Input :
We can select EasyOcr, tesseract or ComputerVision to analyze the image. Upload the image file, if we want the image to be magnified give 'magnification' as 'true' otherwise give 'false'. If we want to fix the rotation of an unknown image file give 'rotationFlag' as 'true' otherwise 'false'. The fields 'portfolio', 'account' and 'exclusion list' are optional. Portfolio and account can be created from

admin portal and in exclusion list we can mention the PII values which we don't want to be analyzed.



POST /rai/v1/privacy/image/analyze Image Analyze

Parameters

Name	Description
ocr string *(query)*	ComputerVision

Request body required

magnification * required
string
False

rotationFlag * required
string
True

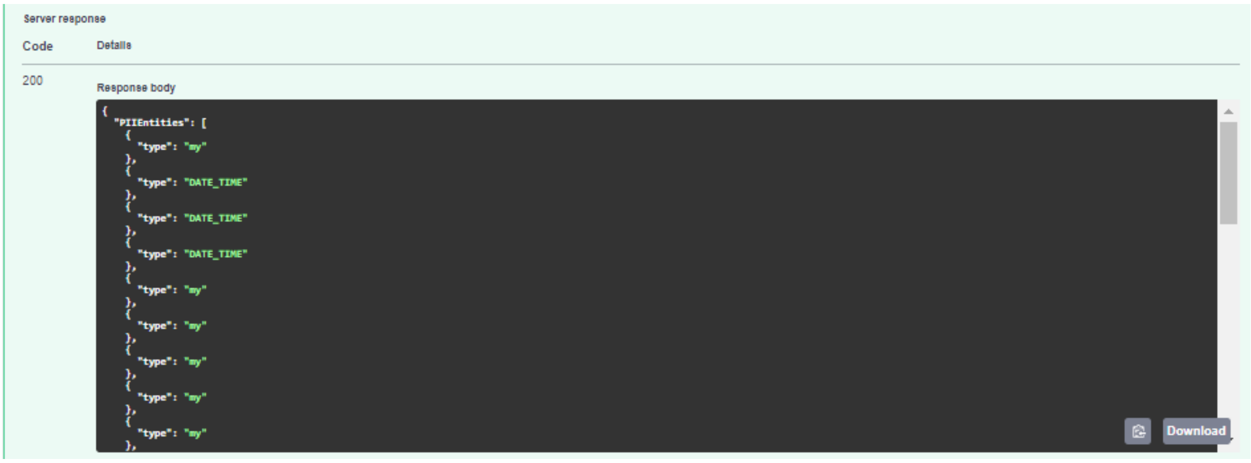
image * required
string($binary)
Choose File BCEP_Level9_result.jpg

portfolio
string
portfolio
☑ Send empty value

account
string
account
☑ Send empty value

exclusionList
string
exclusionList
☑ Send empty value

Response :



Server response

Code Details

200 Response body

```
{
  "PIIEntities": [
    {
      "type": "my"
    },
    {
      "type": "DATE_TIME"
    },
    {
      "type": "DATE_TIME"
    },
    {
      "type": "DATE_TIME"
    },
    {
      "type": "my"
    },
    {
      "type": "my"
    },
    {
      "type": "my"
    },
    {
      "type": "my"
    },
    {
      "type": "my"
    },
```

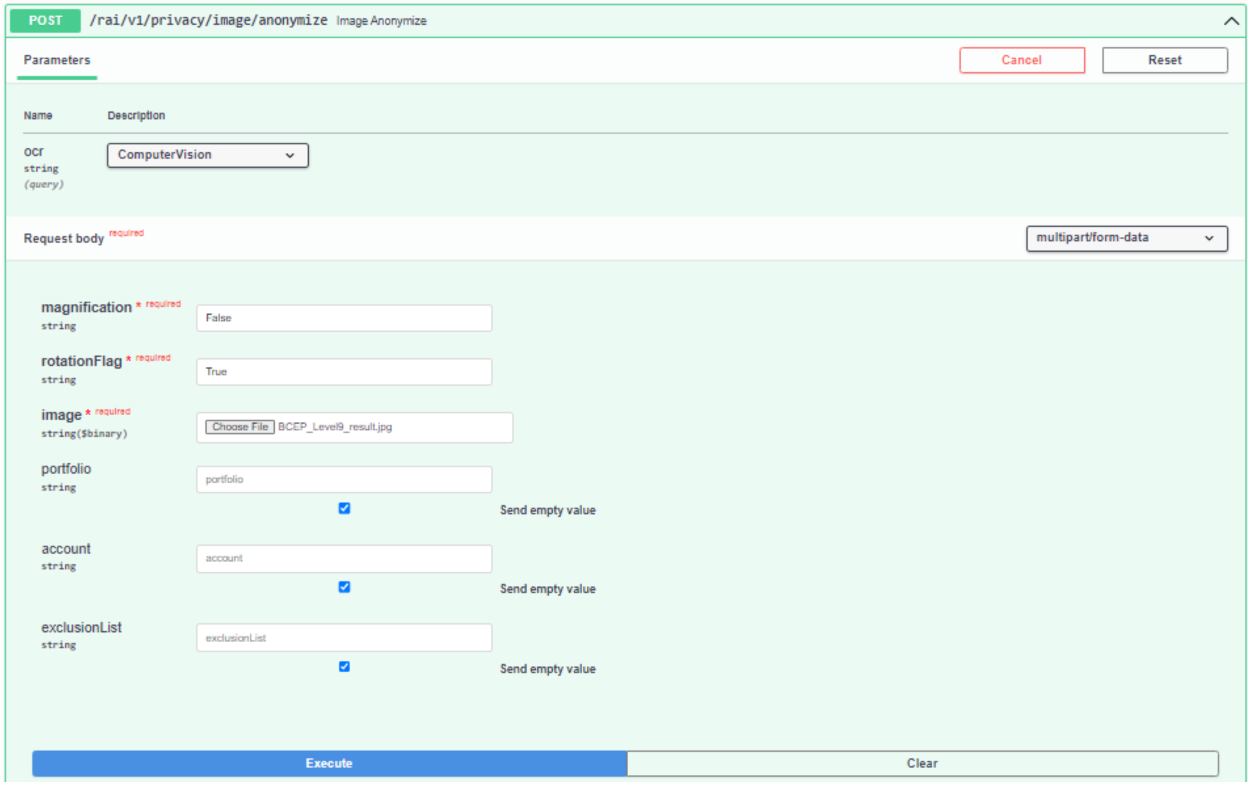
Image Anonymize

Endpoint – /rai/v1/privacy/image/anonymize

Using this API, we can anonymize the PII entities present in the uploaded image.

Input :

We can select EasyOcr, tesseract or ComputerVision to analyze the image. Upload the image file, if we want the image to be magnified give 'magnification' as 'true' otherwise give 'false'. If we want to fix the rotation of an unknown image file give 'rotationFlag' as 'true' otherwise 'false'. The fields 'portfolio', 'account' and 'exclusion list' are optional. Portfolio and account can be created from admin portal and in exclusion list we can mention the PII values which we don't want to be anonymized.



POST /rai/v1/privacy/image/anonymize Image Anonymize

ocr string (query): ComputerVision

Request body required: multipart/form-data

magnification * required string: False

rotationFlag * required string: True

image * required string($binary): Choose File BCEP_Level9_result.jpg

portfolio string: portfolio

☑ Send empty value

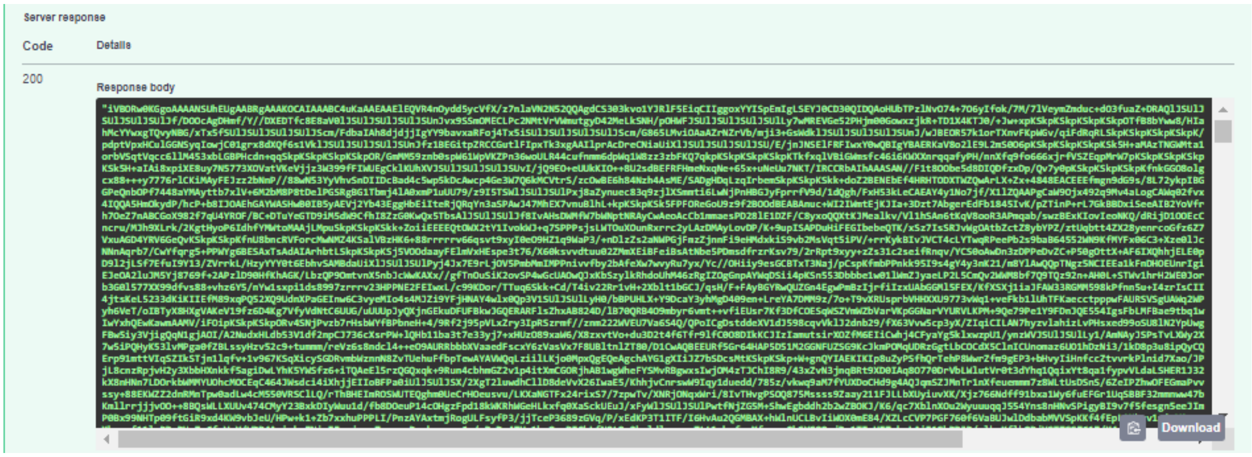
account string: account

☑ Send empty value

exclusionList string: exclusionList

☑ Send empty value

Response :



Server response

Code Details

200 Response body

```
"iVBORw0KGgoAAAANSUhEUgAABRgAAAKOCAIAAABC4uKaAAEAAElEQVR4nOydd5ycVfX/z7nlaVN2N52QQAgdCS303kvo1YJRlF5EiqCIIggoxYYISpEmIgLSEYJ0CD30QIDQAoHUbTPzlNvO74+7O6yIfok/7M/7lVeymZmduc+dO3fuaZ+DRAQlJSUlJ
SUlJSUlJSUlJf/DOOcAgDHmf/Y//DXEDTfc8E8aV0lJSUlJSUlJSUlJSUnJvx9SSmOMECLPc2NMtVrVWmutgyD42MeLkSNH/pOHWFJSUlJSUlJSUlJSUlLy7wMREVGe52PHjm00GowxzjkR+TD1X4KTJ0/+Jw+xpKSkpKSkpKSkpOTfB8bYww8/HIa
hMcYYwxgTQvyNBG/xTx5fSUlJSUlJSUlJSUlJScm/FdbaIAh8djdjjIgYY9bavxaRFoj4TxSiSUlJSUlJSUlJSUlJScm/G865LMviOAaAZrNZrVb/mji3+GsWdklJSUlJSUlJSUlJSUnJ/wJBEOR57k1orTXnvFKpWGv/qiFdRqRLLSkpKSkpKSkpKSkpK/
pdptVpxHCulGGNSyqIowjC01grx8dXQf6s1VklJSUlJSUlJSUlJSUnJfz1BEGitpZRCCGutlFIpxTk3xgAAIlprAcDreCNiaUiXlJSUlJSUlJSUlJSUlJSU/E/jnJNSElFRFIwxY0wQBIgYBAERKaV8o2lE9L2mS0O6pKSkpKSkpKSkpKSk5H+aMAzTNGWMta1
orbVSqtVqcc6llM453xbLGBPHcdn+qqSkpKSkpKSkpKSkpOR/GmMM59znb0spW61WpVKZPn36woULR44cufnmm6dpWq1W8zz3zbFKQ7qkpKSkpKSkpKSkpKTkfxqlVBiGWmsfc46i6ZEqpyPH/nnXfq9fo666xjrfVSZEqpMrW7pKSkpKSkpKSkp
KSkSH+aIAi8xpiXE8uy7N5773XOVatVKeVjjz3W399fFIWUEgCklKUhXVJSUlJSUlJSUlJSUvI/jQ9EO+eUUkKIO++8U2sdBEFRFHmeNxqNe+65x+uNeUu7NKT/IRCCRbAIhAAASAN//F1t8OObe5d8DIQDfzxDp/Qv7y0pKSkpKSkpKSkpKfnkGGO8olg
cx88+++y7776rlCKiMAyFEJzz2bNnP//88wN53YyVhvSnDIIDcBad4c5wp5kDcAwcp4Ge3W7Q6kMCVtr5/zcOwBE6h84Nzh4AsME/SADgHDqLzqIrbemSkpKSkpKSkk+doZ2BENEbEf4HRHTODXTWZQwArLX+Zx+4848EACEEEfmgn9dG9s/8L72ykpIBG
GPeQnbOPf7448aYMAyttb7xlV+6M2bM8P8tDelPGSRg8G1Tbmj4lA0xmP1uUU79/z9I5TSWlJSUlJSUlPxj8aZynuec83q9zjlXSmmti6LwNjPnHBGJyFprrfV9d/1dQgh/FxH53kLeCAEAY4y1No7jf/X1lZQAAPgCaW9Ojx492q9Mv4aLogCAWq02fvx
4IQQA5HmOkydP/hcP+b8IJOAEhGAYWASHw80IBSyAEVj2Yb43EggHbEiIteRjQRqYn3aSPAwJ47MhEX7vnuBlhL+kpKSkpKSk5FPFOReGoU9z9f2BOOdBEABAmuc+WI2IWmtEjKJIa+3Dzt7AbgerEdFb184SIvK/pZTinP+rL7GkBBDxiSeeAIB2YoVfr
h7OeZ7nABCGoX982f7qU4YROF/BC+DTuYeGTD9iM5dW9CfhI8ZzG0KwQx5TbsAlJSUlJSUlJf8IvAHsDWMfW7bWNptNRAyCwAeoAcCb1mmaesPD28lE1DZF/C8yxoQQXtKJMealkv/Vl1hSAn6tKqV8ooR3APmqab/swzBExKIovIeoNKQ/dRijD1OOEcC
ncru/MJh9XLrk/2KgtHyoP6IdhfYMWtoMAAjLMpuSkpKSkpKSkk+ZoiiEEEEQtOWX2tY1IvokWJ+q7SPPPsjsLWTOuXOunRxrrc2yLAzDMAyLovDP/K+9upISAPDuHiFEGIbebeQTK/xSz7IsSRJvWgOAtbZctZ8ybYPZ/ztUqbtt4ZX28yenrcoGfz6Z7
VxuAGD4YRV6GeQvKSkpKSkpKfnU8bncRVForcMwNMZ4KSalVBzHK6+88rrrrv66qsvt9xyI0eO9HZ1q9WaP3/+nDlzZs2aNWPGjFmzZjnnFi9eHMdxkiS9vb2MsVqtSiPV/+rrKykBIvJVCT4cLYTwqRPeePb2s9baB64552WN9KfMYFx06C3+Xze0lJc
NNnAqrb7/CwYfqrg5+PPWYg6BESAxTsAdAIArhbtLSkpKSkpKSj5VDOdaayFElmVxHEspe3t76/X60ksvvdtuu02ZMmXEiBFeiBsAtNbe5PDmsdfrzrKsv79/2rRpt9xyy+zZs31c2seifRqv/YCS0oAwDn3zDPPeDvZC+P50gOttX+AF6IXQhhjELE0p
D9l2jLSf7EfuI9Y13/ZVrrkL/HzyYYY0t6EbhvSAMBdaUiXlJSUlJSUlPyj4Jx7E9rLjOjbVmnMmIMPPnivvfby2bAfeXw7wvyRu7yx/Yc//OHiiy9esGCBTxT3Naj/pCspKfmbPPnkk9SI9s4gY4y3nK21/m8YlAwQQpTNgzSNCIEEa1kFnDHOEUnrIgi
EJeOA2luJM5Yj8769f+2APzlD90HfKhAGK/LbzQP9OmtvnX5nbJcWwKAXx//gfTnOuSiK2ovSP4wGcUAOwQJxKbSzylkRhdoUhM46zRgIZOgGnpAYWqDSii4pKSn553Dbbbe1w01lWmZJyaeLP2L5CmQv2WWM8bf7Q9TQz92n+AH0L+STWv1hrH2WE0Jor
b3G0l577XX99dfvs88+vhz6Y5/nYw1sxpi1ds8997zrrrv23HPPNE2FEIwxL/c99KDor/TTuq6Skk+Cd/T4iv22Rr1vH+2Xblt1bGCJ/qsH/F+FAyBGYRwQUZGn4EgwPmBzIjrfiIzxUAbGGMl5FEX/KfXSXj1iaJFAW33RGMM598kPfnn5u+I4zrIsCII
4jtsKeL5233dKiKIIEfM89xqPQ52XQ9UdnXPaGEInw6C3vyeMIo4s4MJZi9YFjHNAY4wlx0Qp3V1SUlJSUlLyH0/bBPUHLX+Y9DcaY3yhMgD409en+LreYA7DMM9z/7o+T9vXRUsprbVHHHXXU9773vWq1+veFkb1lUhTFKaecctpppwFAURSVSgUAWq2WP
yh6VeT/oIBTyX8HXgVAKeV19fz6D4Kg7VfyVdNtC6UUG/uUUUpJyQXjnGEkuDFUFBkwJGQERARFlsZhxAB824D/lB70QRB4O9mbyr6vmt++vfiEUsr7Kf3DfCOESqWSZVmWZbVarVKpGGNarVYURVLKPM+9Qe79Pe1Y9FDnJQE554IgsFbLMFBae9tbq1w
IwYxhQEwKawmAAMV/iFOipKSkpKSkpORv4SNjPvzb7rHsbWYfBPbneH+4/9Rf2j95pVLxZry3IpRSzrmf//znm222WVEU7Va6S4Q/QPoICgDstddeXV1dJ598cqvVklJ2dnb29/fX63Vvw5cp3yX/ZIqiCILAN7hyzvlahizLvPHsxed99oSU8lN2YpUwg
FBwSiy3VjigQqN1gjAOI/A2NudxHLdb53V1df2npCJ736cXsrPW+lQHb11ba3t7e33yj7+xHUzO89xaW6/X8zxvtVo+du3D2t4f6Tfr9lfC0O8DIkKCJIzIamutsirXOZfM6EIiCwhj4CFyaYgSklxwzpU1/ynzWVJSUlJSUlLy1/AmNAyJSPsTvLXWy2X
7w5iPQHyK53lvMPga0fZBLssyHzv52c9+tummm/reVz6s8ndcl4++eO9AURRbbbXVaaedFscxY6zVasVx7F8UBltnlZT80/D1CwAQBEEURf5Gr64HAP5D51M2GGNFUZSG9KcJkmPOMqUDRzGgtLbCOCdX5ClnICUnomaz6UO1hDzNi3/1kD8p3u8ipQyCQ
Erp91mttVIqSZIkSTjn1lqfv+1v967KSqXicySGDRvmbWznnN8ZvTUehuFfbpTewAYAVWQqLziilLKjo8MpxQgEQeAgchAYG1gXIiJZ7bSDcsMtKSkpKSkp+W+gnQYIAEKIKIp8uZyPSfhQrTehP8WwrZfm9gEP3+bHvyIiHnfccZtvvrkPlnid7Xao/JP
jL8cnzRpjvH2y3XbbHXnkkfSagiDwLYhK5YWSfz6+iTQAeEl5rzQGQxqk+9Run4cbhmGZ2v1p4itXmCGORjhAB1wgWheFYSMvRBgwxsIwjOM4zTJChI8R9/43xZvN3jnqBRt9XD0IAq8O770DrVbLWlutVr0t3dYhq1QqixYt8qa1fypvVLdaLSHER1J32
kX8nHNn7LDOrkbWMMYUOhcMOCEqC464JWsdci4iXhjjEIIoBFPa0iUlJSUlJSX/2XgT2luwdhCllD8deVvX26IwaE5/KhhjvCnrswW9Iqy1duedd/785z/vkwq9aM7fYUXDoCHd9g4AQJqmSZJMnTr1nXfeuemm7z8WLtUsDSn5/6ZeIPZhwOFEGmaPvv
ssy+88EKWZZ2dnRMnTpw0adLw4cM55VRSClLQ/rThBHEImROSWUTEQghm0UeCrHOeusvu/LKXaNGTFx24rixS7/7zpwTv/XNRjONqxWri/8IvTHvgPSOQ875Mssss9Zaay211FJLLbXUyiuvXK/Xjz766Ndff91bxa1Wy6fuEFGr1UqSBBF32mmmww47b
Kmllrrjjjv0O++8BQsWLLXUUv474CMyY23BxkDIyWuu1d/fb8DOeuP14cOHgzFpd18kWKRhWGeHLkxfq0XaSckUEuJ/xFyWlJSUlJSUlPwtfNjZGSM+ShwEgbddh2b2wZB0KJ/K6/qc7XblnXOu2WyuuuuqqJ554Yns8nHNvSPigyBI9v7f5fesgn5eeJIn
P0Bx99NHTp09ftGiR9xd4KW9vbJeU/HPw+k1+Zb7xxhuPPPLI/PnzAYAxtmjRogfULFsyfP3/jjTceP3689zGVq/P/xEdKP3T1ITF/I6HvAu2QGMBAX+hWlnUCLBvIiWOX0mE84/XZLcCVP7PGF760f6VaBUJwlDdbabMVVSpKKf4fEp
```

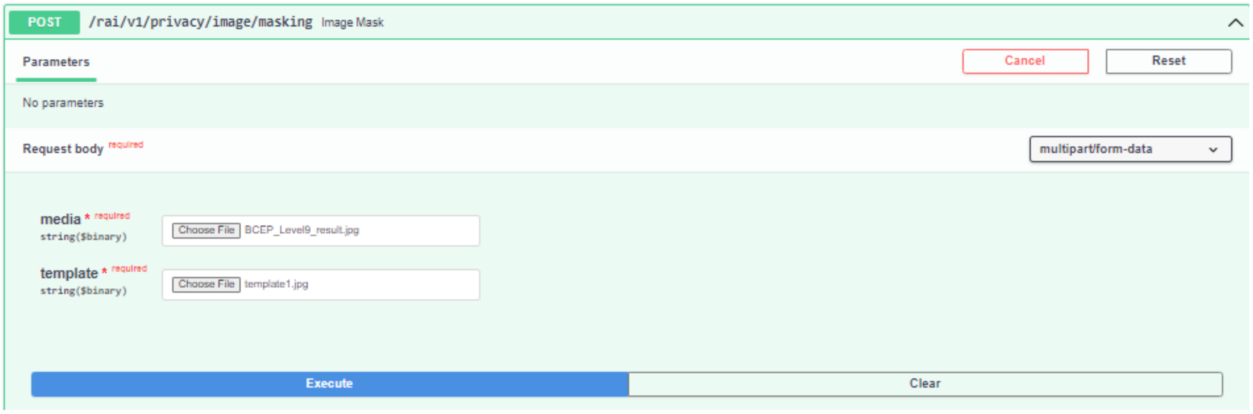
Image Mask

Endpoint – /rai/v1/privacy/image/masking

Using this API, we can mask the uploaded image with the template we uploaded (section of the image).

Input :

Upload the image to be masked and the template(part of the image we want to mask)



POST /rai/v1/privacy/image/masking Image Mask

No parameters

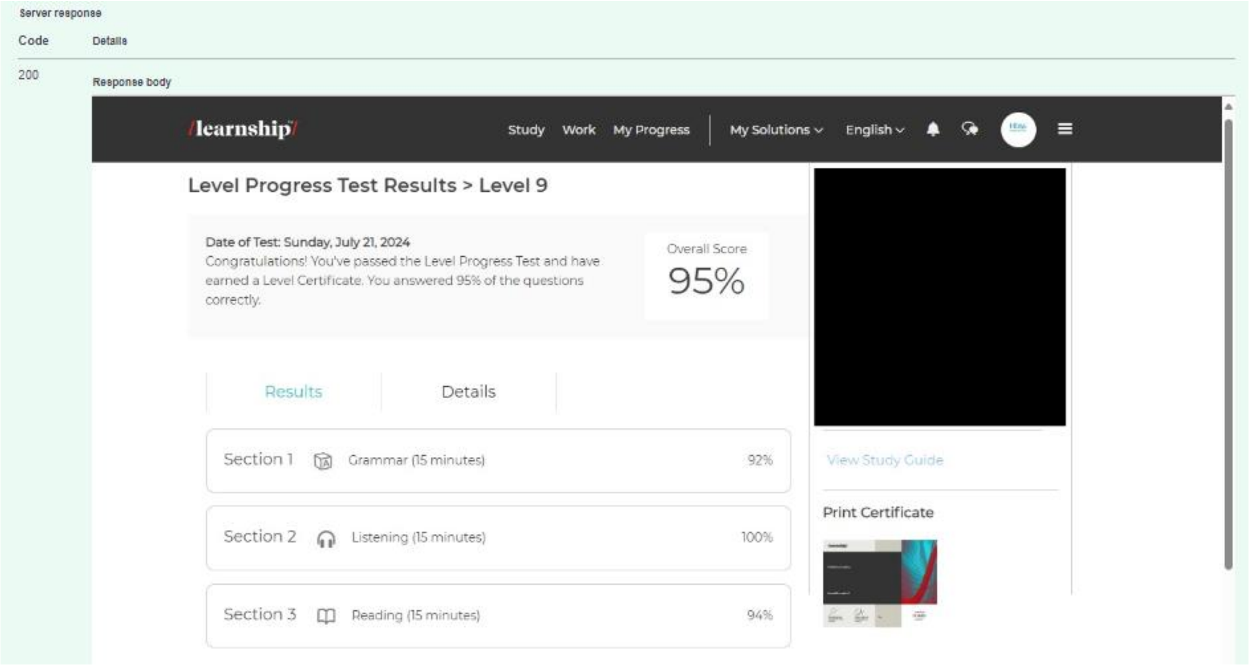
Request body required

multipart/form-data

media * required
string($binary)
Choose File BCEP_Level9_result.jpg

template * required
string($binary)
Choose File template1.jpg

Response :



Server response

Code Details

200 Response body

Level Progress Test Results > Level 9

Date of Test: Sunday, July 21, 2024
Congratulations! You've passed the Level Progress Test and have earned a Level Certificate. You answered 95% of the questions correctly.

Overall Score
95%

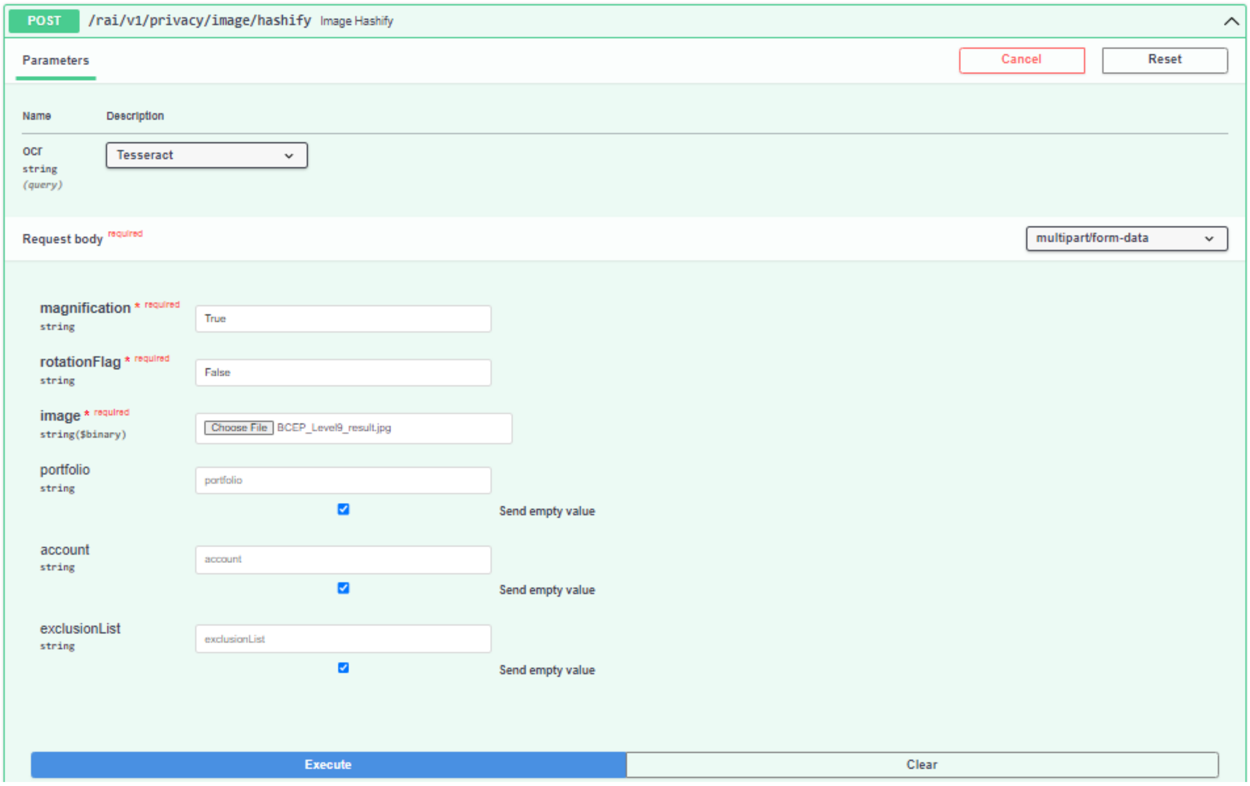
[redacted]

Section 1	Grammar (15 minutes)	92%
Section 2	Listening (15 minutes)	100%
Section 3	Reading (15 minutes)	94%

Image Hashify

Endpoint – /rai/v1/privacy/image
Using this API, we can hash out the PII entities detected in the image we uploaded.

Input :
We can select EasyOcr, tesseract or ComputerVision to analyze the image. Upload the image file, if we want the image to be magnified give ‘magnification’ as ‘true’ otherwise give ‘false’. If we want to fix the rotation of an unknown image file give ‘rotationFlag’ as ‘true’ otherwise ‘false’. The fields ‘portfolio’, ‘account’ and ‘exclusion list’ are optional. Portfolio and account can be created from admin portal and in exclusion list we can mention the PII values which we don’t want to be hashed in the image.



POST /rai/v1/privacy/image/hashify Image Hashify

Parameters

Cancel Reset

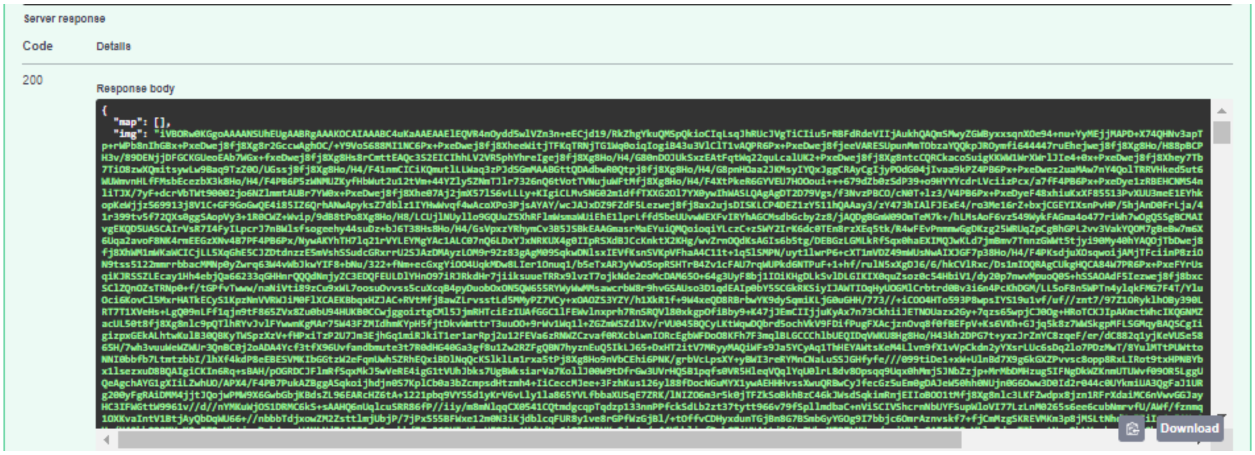
Name	Description
ocr string *(query)*	Tesseract

Request body required multipart/form-data

magnification * required string	True
rotationFlag * required string	False
image * required string($binary)	Choose File BCEP_Level9_result.jpg
portfolio string	portfolio ☑ Send empty value
account string	account ☑ Send empty value
exclusionList string	exclusionList ☑ Send empty value

Execute Clear

Response :



Server response

Code	Details
200	Response body

```
{
  "map": [],
  "img": "iVBORw0KGgoAAAANSUhEUgAABRgAAAKOCAIAAABC4uKaAAEAAElEQVR4nOyddSwlVZn3n+eECjd19/RkZhgYkuQMSpQ[illegible]
```

Download

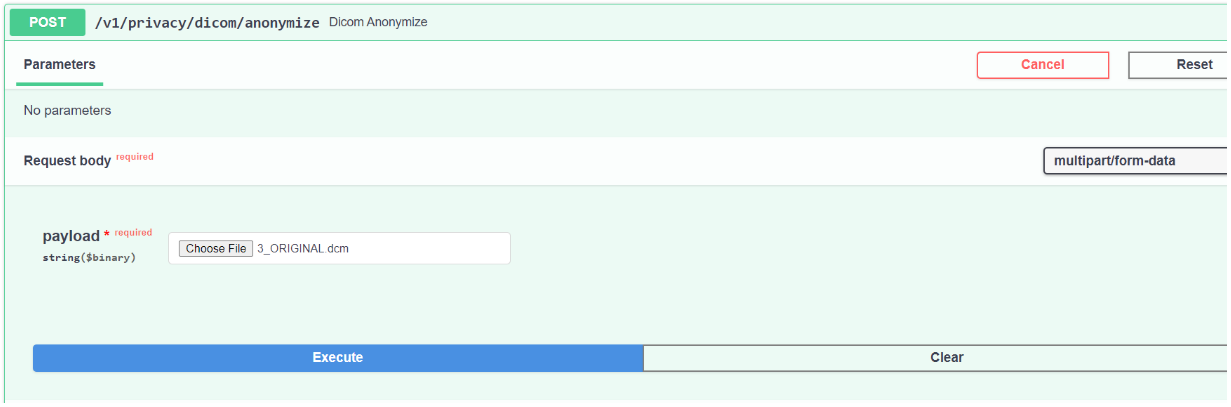
Dicom Anonymize

Endpoint – /rai/v1/privacy/dicom/anonymize

Using this API, we can identify and anonymize the PII entities present in dicom images.

Input :

Upload the .dcm image file



POST /v1/privacy/dicom/anonymize Dicom Anonymize

No parameters

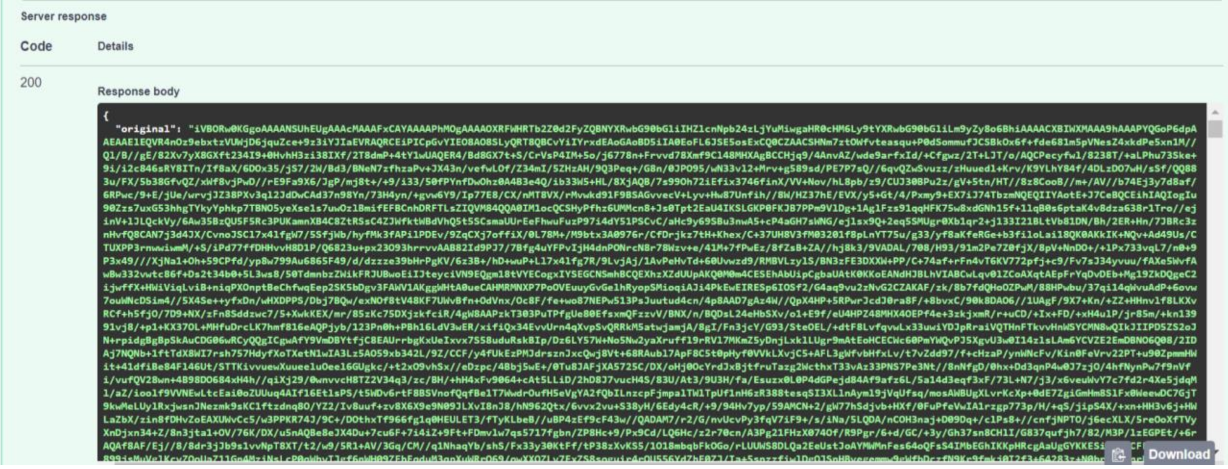
Request body required

multipart/form-data

payload * required
string($binary)

3_ORIGINAL.dcm

Response :



Server response

Code	Details
200	Response body

```
{
  "original": "iVBORw0KGgoAAAANSUhEUgAAAcMAAAFxCAYAAAAPhMOgAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlcnNpb24zLjYuMiwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy8o6BhiAAAACXBIWXMAAA9hAAAPYQGoP6dpA
AEAAE1EQVR4nOz9ebxtzVUWjD6jquZce+9z3iYJIaEVRAQRCEiPICpGvYIEO8AO8SLyQRT8QBCvYiIYrxdEAoGAoBD5iIA0EoFL6JSE5osExCQ0CZAACSHNm7ztOWfvteasqu+P0dSommufJCSBkOx6f+fde681m5pVNesZ4xkdPe5xn1M//
Q1/B//gE/82Xv7yX8GXft234I9+0HvhH3zi38IXf/2T8dmP+4tY1wUAQER4/Bd8GX7t+S/CrVsP4IM+5o/j6778n+Frvvd78Xmf9C148MHXAgBCCHjq9/4AnvAZ/wde9arfxId/+Cfgwz/2T+LJT/o/AQCPecyfwl/8238T/+aLPhu73Ske+
9i/i2c846sRY8ITn/If8aX/6DOx35/jS7/2W/Bd3/BNeN7zfhzaPv+JX43n/vefwLOf/Z34mI/5ZHzAH/9Q3Peq+/G8n/0JPO95/wN33v12+Mrv+g589sd/PE7P7sQ//6qvQZwSvuzz/zHuued1+Krv/K9YLhY84f/4DLzDO7wH/sSf/QQ88
3u/FX/5b38GfvQZ/xWf8vjPwD//rE9Fa9X6/JgP/mj8t+/+9/i33/50fPYnfDwOhz0A4B3e4Q/ib33W5+HL/8XjAQB/7s99Oh72iEfix3746finX/VV+Nov/hL8pb/z9/CUJ30BPu2z/gV+5tn/HT//8z8CooB//m+/AV//b74Ej3y7d8af/
6RPwc/9+E/jUe/wrvjJZ38PXv3ql2JdDwCAd37n98Yn/73H4yn/+gvw6Y9/Ip77E8/CX/nMT8VX/rMvwkd91F9BSAGvvecV+Lyv+Hw87Unfih//8W/HZ37hE/EVX/y5+Gt/4/Pxmy9+EX7iJ74TbzmNQEQIIYAotE+J7CeBQCEihIAQIogIu
90Zzs7uxG53hhgTYkyYphkp7TBNO5yeXsels7uwOzlBmifEFBCnhDRFTLsZIQVM84QQA0IMlocQCSHyPfhz6UMMcn8+Js0Tpt2EaU4IKSLGKP0FKJB7PPm9V1Dg+1AglFzs91qqHFK75w8xdGNh15f+11qB0s6ptaK4v8dza638r1Tro//ej
inV+lJLQckVy/6Aw35BzQU5F5Rc3PUKamnXB4C8ZtRSsC4ZJWfktWBdVhQ5t5SCsmaUUrEeFhwuFuzP97i4dY51PSCvC/aHc9y69SBu3nwA5+cP4aGH7sWNG/ejlsx9Q+2eq5SMUgr0Xb1qr2+j133I21BLtVb81DN/Bh/2ER+Hn/7JBRc3z
nHvfQ8CAN7j3d4JX/CvnoJSCl7x41fgW7/5SfjWb/hyfMk3fAPilPDEv/9ZqCXj7offiX/0L78M+/M9btx3A0976r/CfDrjkz7tH+Khex/C+37UH8V3fM03201f8pLnYT75u/g33/yf8aKfeRGe+b3filoLai18QK0AKkIK+NQv+Ad49Us/C
TUXPP3rnwwiwmM/+S/iPd77ffDHHvvH8D1P/Q6823u+px23O93hrrvvAAB82Id9PJ7/7Bfg4uYFPvIjH4dnPONrcN8r78Wzv+e/41M+7fPwEz/8fZsB+ZA//hj8k3/9VADAL/708/H93/91m2Pe7Z0fjX/8pV+NnDO+/+1Px733vqL7/n0+9
P3x49///XjNa1+Oh+59CPfd/yp8w799Au6865F49/d/dzzze39bHrPgKV/6z3B+/hD+wuP+L17x41fg7R/9LvjAj/1AvPeHvTd+60Uvwzd9/RMBVLzylS/BN3zFE3DXXW+PP/C+74af+rFn4vT6KV772pfj+c9/Fv7sJ34yvuu/fAXe5WvfA
wBw332vwtc86f+Ds2t34b0+5L3ws8/50TdmnbzZWikFRJUBwoEiIJteyciVN9EQgm18tVYECogxIYSEGCNSmhBCQEXhzXZdUUpAKQ0M0m4CESEhAbUipCgbaUAtK0KKoEANdHJBLhVIABCwLqv0lZCoAXqtAEpFrYqDvDEb+Mg19ZkDQgeC2
ijwffX+HWiViqLviB+niqPXOnptBeChfwqEep2SK5bDgv3FAWV1AKggWHtA0ueCAHMRMNXP7PoOVEuuyGvGelhRyopSMioqiAJi4PkEwEIRESp6IOSf2/G4aq9vu2zNvG2CZAKAF/zk/8b7fdQHoOZPwM/88HPwbu/37qi14qWvuAdP+6ovw
7ouWNcDSim4//5X4Se++yfxDn/wHXDPPS/Dbj7BQw/exNOf8tV48KF7UWvBfn+OdVnx/Oc8F/fe+wo87NEPw513PsJuutud4cn/4p8AAD7gAz4W//QpX4HP+5RPwrJcdJ0ra8F/+8bvxC/90k8DAO6//1UAgF/9X7+Kn/+ZZ+HHnvlf8LKXv
RCf+h5fjO/7D9+NX/zFn8Sddzwc7/5+XwkKEX/mr/85zKc75DXjzkfciR/4gW8AAPzkT303PuTPfgUe80EfsxmQFzzvV/BNX/n/BQDsL24eHbSXv/o1+E9f/eU4HPZ48MHX4OEPf4e+3zkjxmR/r+uCD/+Ix+FD/+xH4u1P/jr85m/+kn139
91vj8/+p1+KX37OL+MHfuDrcLK7hmf816eAQPjyb/123Pn0h+PBh16LdV3wER/xifiQx34EvvUrn4qXvpSvQRRkM5atwjamjA/8gI/Fn3jcY/G93/SteOEL/+dtF8LvfqvwLx33uwiYDJpRraiVQTHnFTkvvHnWSYCMN8wQIkJIIPD5ZS2oJ
N+rpidgBgBpSkAuCDG06wRCyQQgICgwAfY9VmDBYtfjC8EAUrrbgKxUeIxvx7S58uduRskBIp/Dz6LY57W+No5Nw2yaXruff19rRV17MKmZ5yDnjLxklLUgr9mAtEoHCECWc60PmYWQvPJ5XgvU3w0I14zlsLAm6YCVZE2EmDBNO6Q08/2ID
Aj7NQNb+1ftTdX8WI7rsh757HdyfXoTXetN1wIA3Lz5AO59xb342L/9Z/CCF/y4fUkEzPMJdrsznJxcQwj8Vt+68RAubl7ApF8C5t0pHyf0VVkLXvjC5+AFL3gWfvbHfxLv/t7vZdd97/f+cHzaP/ynWNcFv/Kin0FeVrv22PT+u90ZpmmHW
it+41dfiBe84F146Ut/STTKivvuewXuueeluOeel6GUgkc/+t2xO9vhSx//eDzpc/48bj5wE+/0Tu8JAFjXA5725C/DX/oHj0OcYrdJxBjtfruTazg2WcthxT33vAz33PNS7Pe3Nt//8nNfgD/0hx+Dd3qnP4w0J7zjO/4hfNynPw7f9nVf
i/vufQV28wn+4B98DO684xH4h//qiXj29/0wnvvcH8TZ2V34q3/zc/BH/+hH4xFv9064+cAt5LLiD/2hD8J7vucH4S/83U/At3/9U3H/fa/Esuzx0L0P4dGPejd84Af9afz6L/5a14d3eqf3xF/73L+N7/j3/x6veuWvY7c7fd2r4Xe5jdqM
1/aZ/ioolf9VVNEwLtcEai0oZUUuq4AIf16EtlsPS/t5WDv6rtF8BSVnofQqfBe1T7WwdrOufH5eVgYA2fQbILnzcpFjmpalTWlTpUf1nH6zR388tesqSI3XLlnAyml9jVqUfsq/mosAWBUgXLvrKcXp+0dE7ZgiGmHm8SlFx0WeewDC7GjT
9kwMeLUylRxjwsnJNezmk9sKC1ftzdnq8O/YZ2/Iv8uuf+zv8X6X9e9N09JLXvI8nJ8/hN962Qtx/6vvx2vu+S38yH/6Edy4cR/+9/94Hv7yp/59AMCN+2/gW77hSdjvb+HXf/0FuPfeVwIA1rzgp773p/H/+qS/jipS4X/+xn+HH3v6j+HW
LaZbX/zin8fDHvZoEAXUWvCc5/w3PPKR74J/9C+/DOthxTf966fg1q0HEULET3/fTyKLbeB//uBP4zEf9cF43w//QADAM7/r2/G/nvUcvPy3fqV7iF9+/s/iNa/5LQDA/nCOH3naj+D09Dq+/c1Ps8+//cnfjNPTO/j6ecXLX/5reOoXfTVy
XnDjxn34+Z/8n3jta1+OV/76K/DX/u5nAQBe8eJX4Du+7cu6F+7i4iZ+9Ft+FDmvlw7qs5717fgbn/ZP8Hc+9/Px9Cd/LQ6Hc/z2r70cn/A3Pg21FHzX074Of/R9Pgr/6+d/GC/+3y/Gh37sn8CH1I/G837qufjh7/82/M3P/1zEGPEt/+6r
AQAf8AF/Ej//8/8dr3jJb9s1vvNpT8XT/t2/w9/5R1+AV/3Gq/CM//q1NhaqYb/shS/Fx33y30KtFf/tP38zXvKS5/1O18mbqbFkOGo/rLUUWS8DLQa2EeUstJoAYMWMnFes64oQFsS4IMbEGhIKKpHRcgAaUgGYKKESi
```

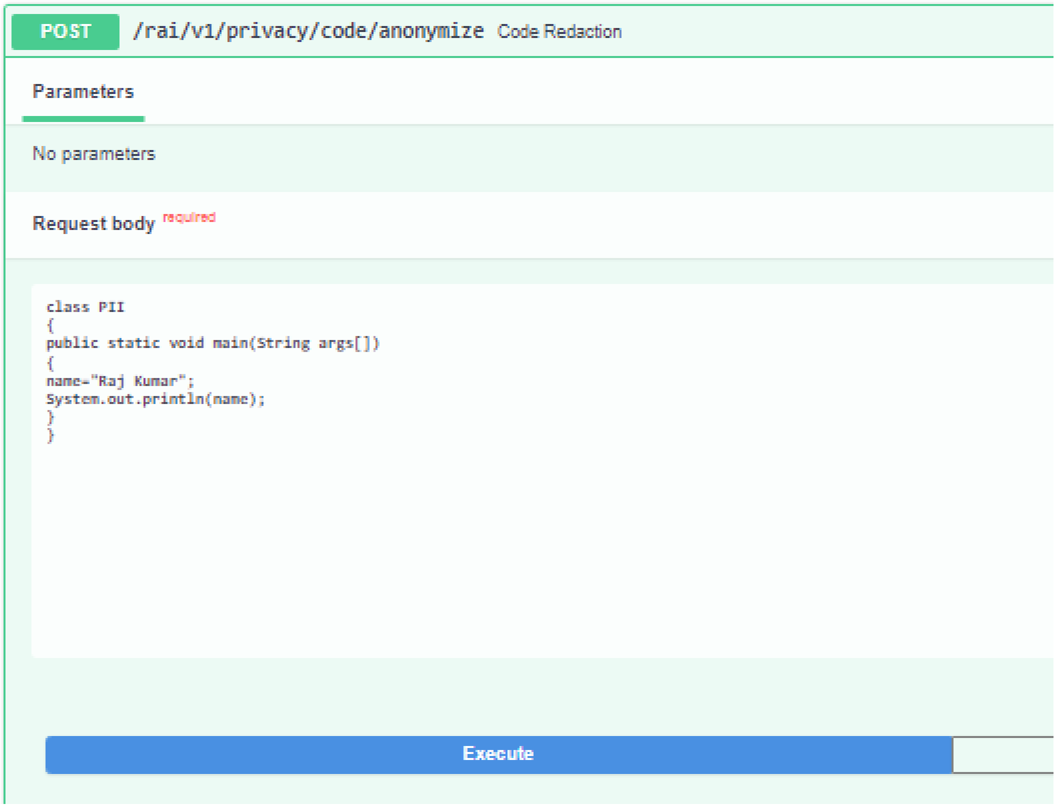
Code Anonymize

Endpoint – /rai/v1/privacy/code/anonymize

Using this API, we can identify and anonymize the PII entities present in the code that we entered as text.

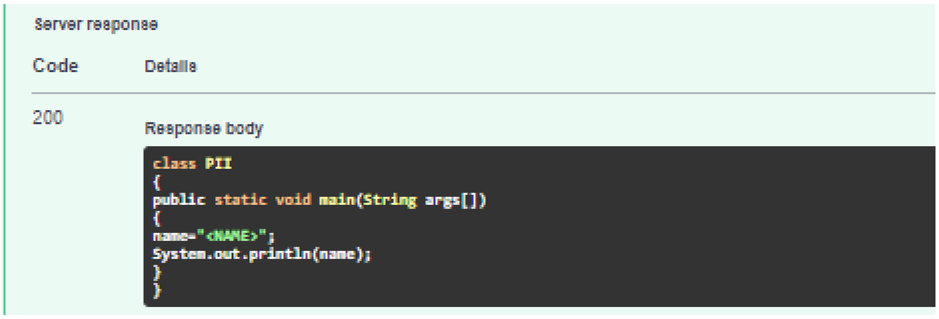
Input :

Enter the code that we want to check for PII entities.



```
class PII
{
public static void main(String args[])
{
name="Raj Kumar";
System.out.println(name);
}
}
```

Response :

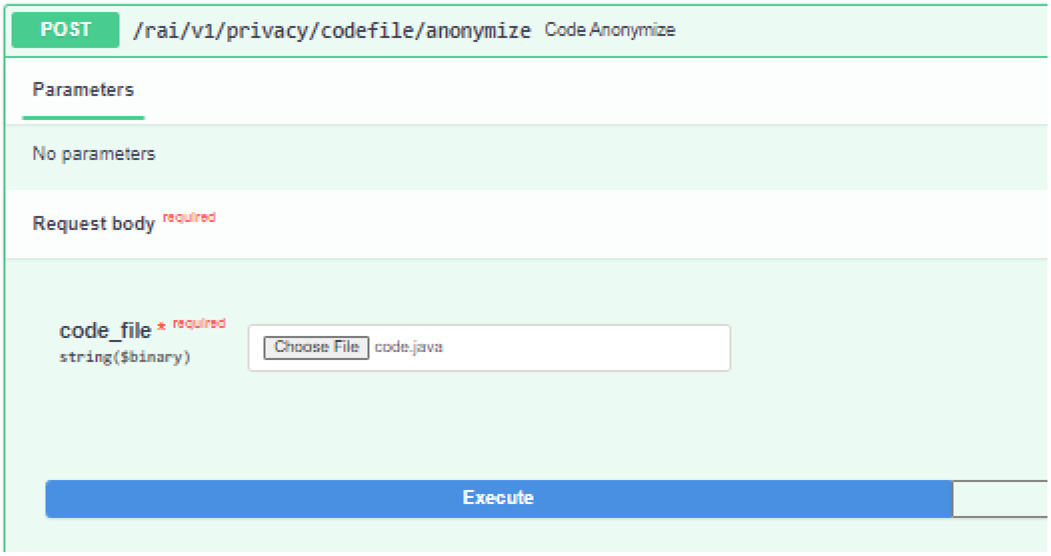


```
class PII
{
public static void main(String args[])
{
name="<NAME>";
System.out.println(name);
}
}
```

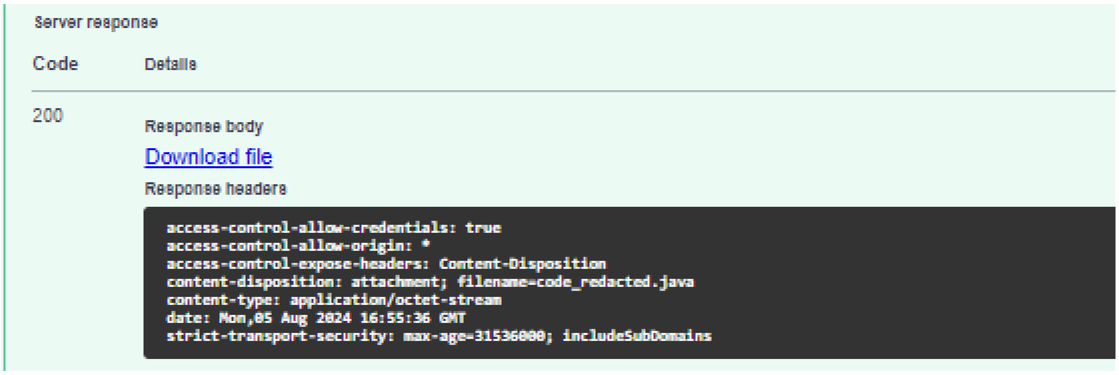
Code File Anonymize

Endpoint – /rai/v1/privacy/codefile/anonymize
Using this API, we can identify and anonymize the PII entities present in the code file that we uploaded as input

Input :
Upload the code file in which you want to find and anonymize the PII entities.



Response :



```
access-control-allow-credentials: true
access-control-allow-origin: *
access-control-expose-headers: Content-Disposition
content-disposition: attachment; filename=code_redacted.java
content-type: application/octet-stream
date: Mon,05 Aug 2024 16:55:36 GMT
strict-transport-security: max-age=31536000; includeSubDomains
```

Downloaded file –

```
class PII
{
public static void main(String args[])
{
name="<NAME>";
System.out.println(name);
}
}
```

Differential Privacy

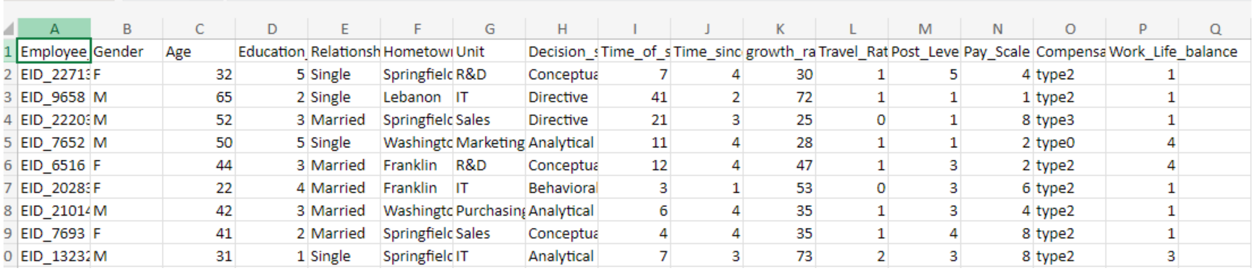
Endpoints – /rai/v1/privacy/DifferentialPrivacy/file

/rai/v1/privacy/DifferentialPrivacy/anonymize

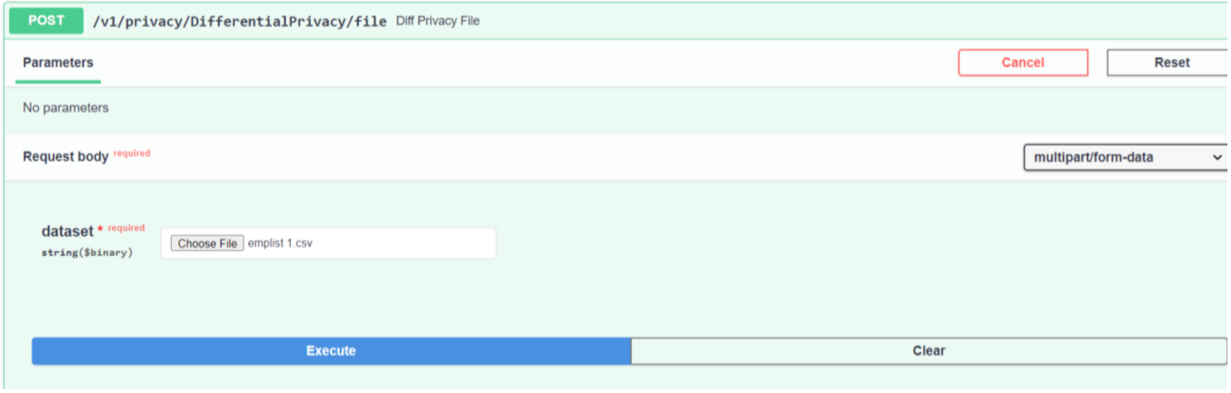
Using the first API, we can upload the file we want to check for differential privacy and using the second API we can add suppression, noise etc. to the file values.

Input :

Upload the file you want to check for differential privacy (using /rai/v1/privacy/DifferentialPrivacy/file) . Example – Here uploading a .csv file-



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Employee	Gender	Age	Education_	Relationsh	Hometowi	Unit	Decision_s	Time_of_s	Time_sinc	growth_ra	Travel_Rat	Post_Leve	Pay_Scale	Compensa	Work_Life_balance	
2	EID_2271:	F	32	5	Single	Springfielc	R&D	Conceptua	7	4	30	1	5	4	type2	1	
3	EID_9658	M	65	2	Single	Lebanon	IT	Directive	41	2	72	1	1	1	type2	1	
4	EID_2220:	M	52	3	Married	Springfielc	Sales	Directive	21	3	25	0	1	8	type3	1	
5	EID_7652	M	50	5	Single	Washingtc	Marketing	Analytical	11	4	28	1	1	2	type0	4	
6	EID_6516	F	44	3	Married	Franklin	R&D	Conceptua	12	4	47	1	3	2	type2	4	
7	EID_2028:	F	22	4	Married	Franklin	IT	Behavioral	3	1	53	0	3	6	type2	1	
8	EID_2101·	M	42	3	Married	Washingtc	Purchasing	Analytical	6	4	35	1	3	4	type2	1	
9	EID_7693	F	41	2	Married	Springfielc	Sales	Conceptua	4	4	35	1	4	8	type2	1	
0	EID_1323:	M	31	1	Single	Springfielc	IT	Analytical	7	3	73	2	3	8	type2	3	



POST /v1/privacy/DifferentialPrivacy/file Diff Privacy File

Parameters

No parameters

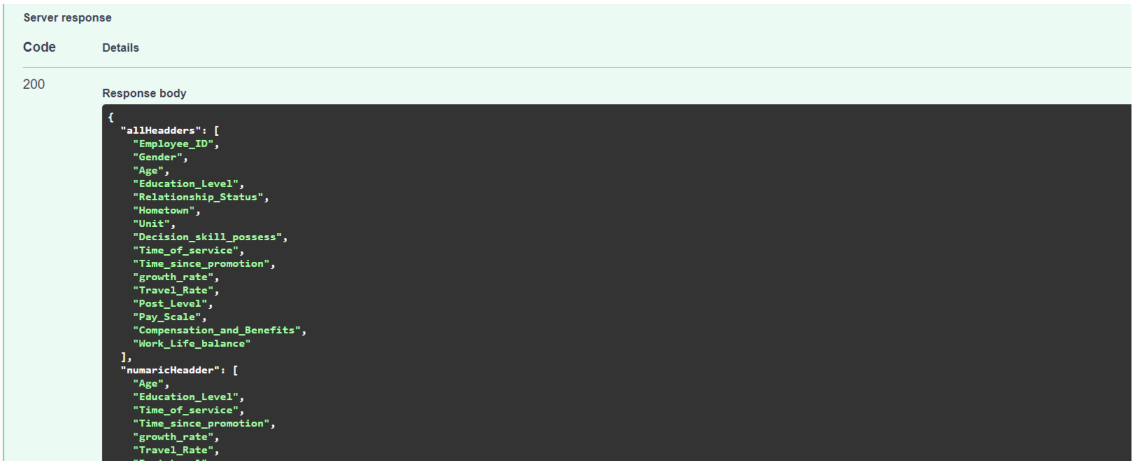
Request body required

multipart/form-data

dataset * required
string($binary)

Choose File emplist 1.csv

Response for file upload:



Server response

Code Details

200

Response body

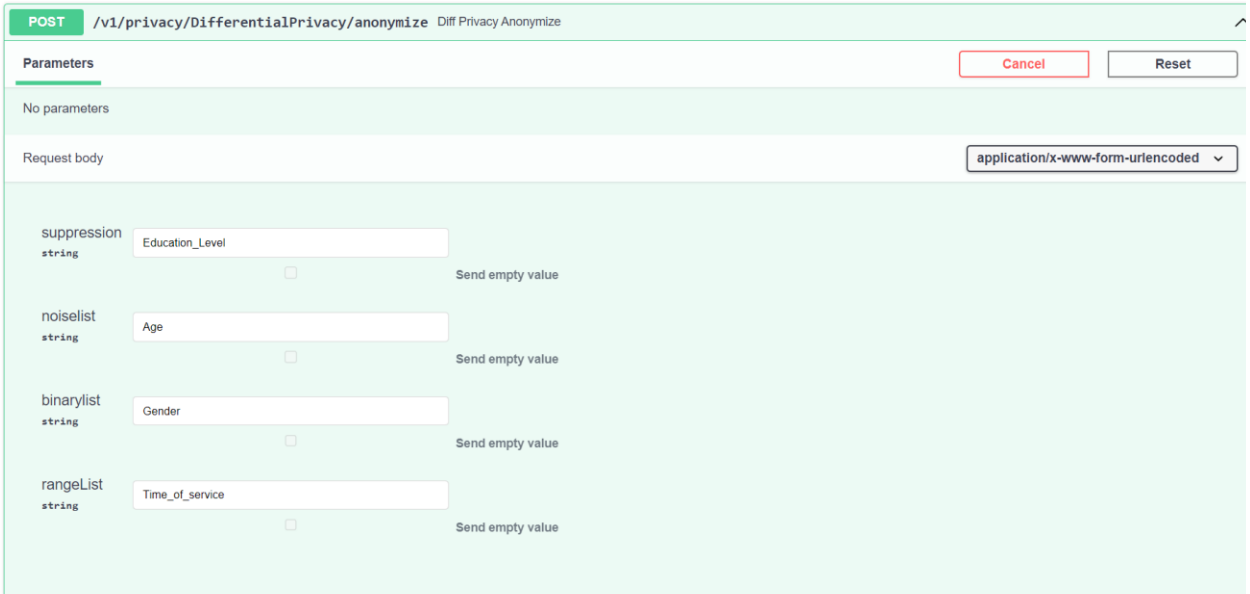
```
{
  "allHeadders": [
    "Employee_ID",
    "Gender",
    "Age",
    "Education_Level",
    "Relationship_Status",
    "Hometown",
    "Unit",
    "Decision_skill_possess",
    "Time_of_service",
    "Time_since_promotion",
    "growth_rate",
    "Travel_Rate",
    "Post_Level",
    "Pay_Scale",
    "Compensation_and_Benefits",
    "Work_Life_balance"
  ],
  "numaricHeadder": [
    "Age",
    "Education_Level",
    "Time_of_service",
    "Time_since_promotion",
    "growth_rate",
    "Travel_Rate",
```

Set the column names –

1. In suppression give the column names which you want to remove or suppress
2. In noise list give the column names whose values you want to change by adding some noise or unwanted

3. In binary list give the column names which contain binary values(only two types of values like M or F, T or F) and whose values you want to anonymize by swapping the binary values at all rows. Example: If a column consists of M and F values, then replace M with F and F with M at all places.
4. In range list add the columns whose values you want to anonymize by converting them to a range.

   Using the second API, it will do these changes on the file uploaded using the first API –



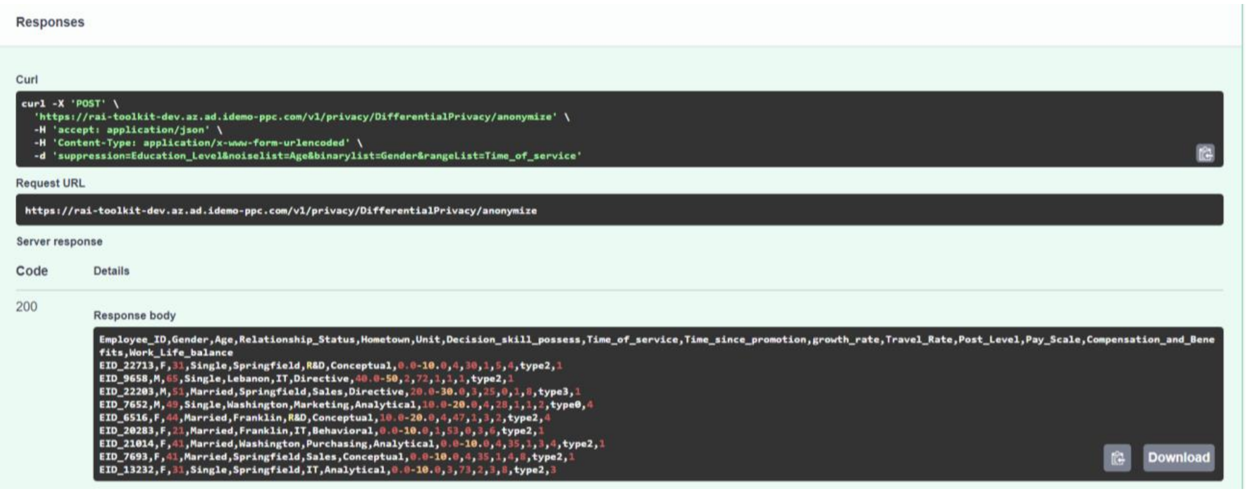
suppression string: Education_Level

noiselist string: Age

binarylist string: Gender

rangeList string: Time_of_service

Final Response :



Curl

```
curl -X 'POST' \
  'https://rai-toolkit-dev.az.ad.idemo-ppc.com/v1/privacy/DifferentialPrivacy/anonymize' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/x-www-form-urlencoded' \
  -d 'suppression=Education_Level&noiselist=Age&binarylist=Gender&rangeList=Time_of_service'
```

Request URL

```
https://rai-toolkit-dev.az.ad.idemo-ppc.com/v1/privacy/DifferentialPrivacy/anonymize
```

200

Response body

```
Employee_ID,Gender,Age,Relationship_Status,Hometown,Unit,Decision_skill_possess,Time_of_service,Time_since_promotion,growth_rate,Travel_Rate,Post_Level,Pay_Scale,Compensation_and_Bene
fits,Work_Life_balance
EID_22713,F,31,Single,Springfield,R&D,Conceptual,0.0-10.0,4,30,1,5,4,type2,1
EID_9658,M,65,Single,Lebanon,IT,Directive,40.0-50,2,72,1,1,1,type2,1
EID_22203,M,51,Married,Springfield,Sales,Directive,20.0-30.0,3,25,0,1,8,type3,1
EID_7652,M,49,Single,Washington,Marketing,Analytical,10.0-20.0,4,28,1,1,2,type0,4
EID_6516,F,44,Married,Franklin,R&D,Conceptual,10.0-20.0,4,47,1,3,2,type2,4
EID_20283,F,21,Married,Franklin,IT,Behavioral,0.0-10.0,1,53,0,3,6,type2,1
EID_21014,F,41,Married,Washington,Purchasing,Analytical,0.0-10.0,4,35,1,3,4,type2,1
EID_7693,F,41,Married,Springfield,Sales,Conceptual,0.0-10.0,4,35,1,4,8,type2,1
EID_13232,F,31,Single,Springfield,IT,Analytical,0.0-10.0,3,73,2,3,8,type2,3
```