**API Documentation**

The SimplyOnline's API is a JSON-based API. All requests are made to endpoints beginning

with: 'http://localhost:3001/'. Requests can be made with http.

After logging in to access APIs that require authentication, please include the generated

token in the header with the key "token."

**1. createService**

This API endpoint creates a new service as specified by the end-user. This is a POST

request. This is sent with the default "Content-Type" header of "application/x-www-form

urlencoded".

*Request type:* **POST**

*Input body type:* **JSON Object**

*Output type:* **JSON Object**

*Sample request:* http://localhost:3001/login

*Sample input:*

{

"mail": "testing@gmail.com",

"password": "Password@123",

}

*Sample output:*

{

"token":

"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyX2ZvdW5kIjoxLCJlbWFpbCI6InRlc3Rp

bmdAZ21haWwuY29tIiwidXNlcm5hbWUiOiJ0ZXN0aW5nIiwidXNlcl9pZCI6MywidGltZSI6Ildl

ZCBPY3QgMDQgMjAyMyAxMToxMToxNCBHTVQtMDQwMCAoRWFzdGVybiBEYXlsaWdo

dCBUaW1lKSIsImlhdCI6MTY5NjQzMjI3NH0.hZImVsqovmUqks9GKzl\_Jkpw-gy

Q6hc3oWRyS4BCSg"

}

*Request type:* **POST**

*Input body type:* **JSON Object**

*Output type:* **JSON Object**

*Sample request:* http://localhost:3001/createUser*Sample input:*

{

"email": "testing@gmail.com",

"username":'test'

"password": "Password@123",

"confirmpassword": "Password@123"

}

*Sample output:*

1. {

"errorMessage": "Email already exists"

}

2. {

"fieldCount": 0,

"affectedRows": 1,

"insertId": 0,

"info": "",

"serverStatus": 34,

"warningStatus": 0

}

*Request type:* **POST**

*Input body type:* **JSON Object**

*Output type:* **JSON Object**

*Sample request:* http://localhost:3001/createRoom

*Sample input:*

{

"room\_name": "testing",

"user\_name": "ajay",

}

*Sample output:*[

[

{

"message": "Room created successfully."

}

],

{

"fieldCount": 0,

"affectedRows": 0,

"insertId": 0,

"info": "",

"serverStatus": 34,

"warningStatus": 0

}

]

*Request type:* **POST**

*Input body type:* **JSON Object**

*Output type:* **JSON Object**

*Sample request:* http://localhost:3001/joinRoom

*Sample input:*

{

"room\_name": "testing11",

}

\*Sample output

[

[

{

"room\_id": 100019,

"owner\_name": "ajay",

"room\_name": "testing11",

"room\_description": "",

"start\_time": "2023-05-03T19:27:16.000Z",

"end\_time": "2023-05-04T19:27:16.000Z",

"max\_capacity": 100,

"current\_capacity": 0,

"room\_password": "",

"is\_locked": 0,

"is\_public": 1,

"room\_type": "","is\_valid": 1

}

],

{

"fieldCount": 0,

"affectedRows": 0,

"insertId": 0,

"info": "",

"serverStatus": 34,

"warningStatus": 0

}

]

*Request type:* **POST**

*Input body type:* **JSON Object**

*Output type:* **JSON Object**

*Sample request:* http://localhost:3001/startAttendance

*Sample input:*

{

"owner\_name": "ajay"

"room\_name": "testing11",

}

\*Sample outpu

[

[

{

"id": 100044,

"owner\_name": "ajay",

"room\_name": "testing11",

"created": "2023-05-03T20:11:56.000Z"

}

],

{

"fieldCount": 0,

"affectedRows": 0,

"insertId": 0,

"info": "",

"serverStatus": 2,

"warningStatus": 0}

]

*Request type:* **POST**

*Input body type:* **JSON Object**

*Output type:* **JSON Object**

*Sample request:* http://localhost:5001/verify

*Sample input:*

{

{

"data\_url":"

T0ZJTEUAAQEAAAHIAAAAAAQwAABtbnRyUkdCIFhZWiAH4AABAAEAAAAAAABhY3Nw

AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAQAA9tYAAQAAAADTLQAAAAAAAA

AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAlk

ZXNjAAAA8AAAACRyWFlaAAABFAAAABRnWFlaAAABKAAAABRiWFlaAAABPAAAABR3d

HB0AAABUAAAABRyVFJDAAABZAAAAChnVFJDAAABZAAAAChiVFJDAAABZAAAAChjc

HJ0AAABjAAAADxtbHVjAAAAAAAAAAEAAAAMZW5VUwAAAAgAAAAcAHMAUgBHAEJY

WVogAAAAAAAAb6IAADj1AAADkFhZWiAAAAAAAABimQAAt4UAABjaWFlaIAAAAAAAAC

SgAAAPhAAAts9YWVogAAAAAAAA9tYAAQAAAADTLXBhcmEAAAAAAAQAAAACZmYAA

PKnAAANWQAAE9AAAApbAAAAAAAAAABtbHVjAAAAAAAAAAEAAAAMZW5VUwAAACA

AAAAcAEcAbwBvAGcAbABlACAASQBuAGMALgAgADIAMAAxADb/2wBDAAMCAgICAgM

CAgIDAwMDBAYEBAQEBAgGBgUGCQgKCgkICQkKDA8MCgsOCwkJDRENDg8QEBEQC

gwSExIQEw8QEBD/2wBDAQMDAwQDBAgEBAgQCwkLEBAQEBAQEBAQEBAQEBAQEB

AQEBAQEBAQEBAQEBAQEBAQEBAQEBAQEBAQEBAQEBAQEBD/wAARCAEsASwDA

SIAAhEBAxEB/8QAHQAAAQQDAQEAAAAAAAAAAAAAAwECBAUABgcICf/EAEAQAAED

AgUCBAUCAwcDAwUAAAEAAgMEEQUGEiExB0ETUWFxCBQigZEyoSOxwRUWJEJS0eE

zQ2IlcoKSosLw8f/EABoBAAEFAQAAAAAAAAAAAAAAAAIAAQMEBQb/xAArEQACAgEEA

gEDAwUBAAAAAAAAAQIDEQQSITEFQRMiMlEUYXEVI0KBkTP/2gAMAwEAAhEDEQA/A

PoNLU3vfsoE9U0X3UWetttf91AqK0cXuVdbK+MhZqoX53JUGaYE8oMtRc3QDJc+iiZJgc94

52QnSA/UhveOb7eSG6QWt2QhxQsru47lR5CbWBSyVEY2UOaquNkiRBnG3khvc229hZQX

VEovY/umCWWQ7HUfJoJQPHYXZJfKOxCDJM0J0dFVy/8AacPV5sjNwWd1tUzQfJrb/uVW

nqqa/ukGqZy6RXumJ/SChGRxNrgfdXrMEiH69T/dxRo8NhYfoga32Cpy8rVH7VkmjpJeyiiiqH

Ou1jyPPTYfupXyMrx9X81ctpQNg38p/wAttbTsqNvk7J/asE8dPGJTx4c0DdHZRRtFmt3Vm2n

BtdqeILDjf2VGd9k/uZYhGMeitFME4UrRv3Vj4PeyURDsFC3kkTIIp2gW33SmmF/pNrKd4Q4

WeHv/AMIOgt2SD8u3unNiF9lL8Ik78LPCIHAQNMRFMJ4CwRHyUoxAd/usMQ7Id3ofBG8I+

acIduBuj+GAQClDPLdDhC5I/hWNjYrPB7lSNN9kmlNwOyPot29lmncC3KO5pCbb0RA9AxG

bfpWBuk2IG6MOLLC0EWT8LoQJ25uE4A2J7JdNjul9EuxZGBu90v1Da6cAVgBSTG4ZOlqX

OJAUZ0j3b9kx000h/hxu39EP5OrfuTYLv5aiuKy2c4q5PhIe57ALl2/uo0tUxrSAd1Kbher/AKjyf

2TxhkDeYwfcKvLyFK6JVp5spzUyP/6Y1e26QR1suwjc0Hz2V82iY39LQLfZOFMB2VSzybfEI

k0dMvbKBuGVDt3yNb7XKIzB2cufI772/kr3wB5J3gX3Cp2a26f+RYjTBLopY8GpGm5gDj/5Xd

/NS46RkYDWxgewsrAQDunCFvlZVZ32S7ZLGMY9IhCAD/KnCIBTDDYXCQRC+6rvL7DyR

RF3us8Lzapfht80gYO6BxwJMjeGPJO8NvFrKQWC9rfhL4fkmwORvDsOLrDGbDayOGWOwWWv7oWIj6TwQkDfQqV4bSmGN3Nk2A0BtvsEmn0R2s91hjAvskPlkfSd9lhCNpFk0sFkhZB

AX4WafMI2m3kkLB5cqKUcjqQLSNrpA3fYImny7JDso3FoLORCB9kwsbyUXlIePVPw+x8g9I

9017QDdELQO+6btbtvylgHIPcbgcrA08kohDeewTTY+yXQkxC0W3O6YdkX1/mmu54ui6E2

N44CQ3TvpIuk280P8jovPlGg7t28k75dvFuFYmEHgWTfDA2O91t4b7M5EDwBbdvCzwWnh

THRG9wNkmmzeBeybaEnghOisP0pror7jZTLEGxCaYydxZJoRF0DyCzw/JSAwEJCwj7oHHI

aYANNuFmgHsjaQQk0oNmB8giyw3Wadroth3SaRbZC4/gdcgdBPASabdlIDRwU0t7BA0PgF

p4CS1+6LpJCzT3QSWBIE5t+AkLDe6MWn1SaUOAgZGxWEcJ2nlJp34SHTwMLXE9k0sub

goh4WWCbI+4EW8kJtrm5CkaQdx3TCw+yGWUElkFbv9klr9kQxkrNB3uE3YumCIsmEb7hH

Mfkmlp7oGmGCA72WFP090mkoUIG5oJumm1uLeSKWppF+R7IZCwCcNk3SQCSilvdNLbnj

dLoWBt/Psmv33TrWFgCfumkGxKJMTG324WC/kkv6JfsUWBuTeHRkeoTHMBPFlKeNroWk

9z+V0G0ysgLJjo+4RnNtvdNQ7R8gCwHsLpumxsUa29wmuv5IWglJgHt2uNkwtBbzupBbshh

u/CBoMFb0WFtxZPc08hIAgYlwDt+Vjm8AJ9t90pHYBRskTBbj2S6du5KfpA2WW7IcBZBlhS6

AQnhpuU6wBsha/IyAafRYW+SKWXShne6jYYAttvZIW3UgM9EwtJNt7IBwHh7JfDF7WR2x3

PKd4fsm2j4Iui3CXwyVJMeyGAC626WQkmgPh7JCze6kAX7LPD+6D+AiMWHlNLD5KT4djz

ZNcy3qmYSIxjNtk3w7dlIMe2wTT/JM0OR3MN9gENzDe9rqU5txsEN7CdzsAgYzI5HmEzTt6

o5btshkW7BDgSYIhNIKe5p5SJ08jg735CwOFlj78gIZJvbUUt2BHQXC4THIzuEK37rpmY4KT

gbJgHYp8nPKbuORfyQsdA3Adk0i4sE4/q2SOOyBoLPAzlNcHXuQnAJHNvwdkLHTY0i42O6

YQDayJ/lumd+FFJBoa4bWTUQ8EBNtvugCRlrb24WNH5S22Siw2QsIQtKxzTdEtdJYqOXI6

GBpG/Kz3CJpJFhykDPNRNYDG6UhYAFRZzz1lfIWFSYvmfE46Wnj7EgvefJreXH0C8350+

OChY6ShyNk+rrXNv/ABq2VsYA89DbuI9yFYp0luoWYR4AndCv7ng9WtbZPDQeF87n/Gr1e

oK19WKrCmMJ2pzTkx28t3X/AHVLifxu9YqmodPT49DS3NwyCkjLR6fUCfyVdj4bUP8ABF+tr

PpXYeYQtAEtgN7XXz1yL8d/UDDcTD84OZilHJs4thZHIz1bazfyD9l2bJXxq5dxLEqiXH5m/Jy

SAQMiYGyRxju4XIJ7mxUNvitTX/jkljq6p+z1K5paRsSsADuCuPT/ABX9HIiwuxmqBkcGN/wrr

G/pzb1st3yx1Jyhmkl2CY9R1IfYhjJQXAeZHIVKemuhzKLSJVbBvCZtBYmuHa3CIx8b/wBD2

u9jdKW3P9FWk8Ei56AOZduyG5u6O5hubJulDu9BYAOaeya5m10ctuChuYSOUw5Hka7cgo

DhzybKU4X3PZR3N3KHsbAEgprveyI4WCG4G+4TJDoE704QnFgNjdGcDff+aEWm/F0sBH

Q3HlCcbbd05xBKFI61wF1DRjYGE373TS4LHWHLv3Qy4X4sgaEhe6R1uyQOuLpt/NA3gcVI

SL7JCbbBYCml+wUULp/ZM2PaycT/ALpoKjYYtjYJCwA78pTvtbhYAbAdkD/YdCtaCl028ko2

Cz3QSQaMATg0d1gFkKqqYaOB9RPI1jGDU5znWACjaYSQtTUU9HC+oqZmRRRtLnOe6w

AHclefurPxfZSyZSvpcrQ/2xXvu2J4JEQI7+b/ALflcz+Iz4qqzEZq7JmRnRtooNdNV1Fw81Dr2I

YRsGjcd7+my874ZlinNA/H8dxllHBUMM5YARNJvYgk2s2/e5v/AC6DQ+H+SPyXr/Rn6nWOL

2V/9Nj6n9acw9S6qGfN04gmgYfAoqOOwZc/qI1Gzt+XLlFfXtpjahhpHN1ElzpHPlHuRt+5TMR

zB4hbQ5awh87HbARxFxeb9yBe3otdr6PEXs8evfBQMcSBG8Oa4H8W/dbtdMaoquCwjLnZK

ct0nyNqq6QSOmqRG8g7NBCrTiDZHFzIY2Ef6jypT8uVA8J8kjaiKZuoOgeC4fYqFXx0MDhH

C+ZjgLfWNwfUKRJZGzwOjq3P1t0saXbjVsD7I2mB0TZInPpJgPq1Ouxx8wVFFJNUsb8zNC

Y+A8O4+yjytFHKaOplIBH0vDrtIT4XoJZXZs1JjmMUzGCaSSYM40v2Wx4L1sxLA6qGnocO

qhVMI0zsdpe32Hf8rlMmI1eGv8GVviUx/S5p2Ct8LnoqhzfFncI3EWlabmM+o8lFZWmuUSRm

1yequmvxW9QMErI5MTxOpq6Nsn101WA6Rze+l5uR+bL3V086i4Fn/AoMZwuqD2TMBLbW

cw23BHZfInA4qhldUMEpmpouZWXDT6ruPw39dca6eZwbhZqJJcFq3/4qmtqttbW0eY9OQs

DyPi43Qcqlhov6bVuL2zfB9NCAR7phaBfZRcJxiixaihq6SYSNljDwR3BCmGxauRlFrhm0uQJ

Btxbuhv2PkjuabXuhvtza6H0GiM4eqG5l+UdwTHBJCZGezf8AqgvHI5UiQE7IEgttfZFjDElwBf

so7zZxsApDjygll/JJ4Yy4N9JFt0Mna6xzjqPbZMe+wtYLpjIQJ5ubd1Do6ySsa/XRT0+h5aPFFi

4A7H78qWdhwmE3QZwPgwkbDssuBtZNvp3ukLroGx8C31HcJb2TbgblJqHmgfISWBxIukJTXHZIDc3PZRsIIHXG/dObyhjdPDrFAx0P4CwJAdk4IQxHvEYuQbLzn8VXV3EMv4dDkbLsY/t

HGGuZqsHvaw2GzRcgkmw23sbL0ftb6uO68JdasKqs09YsYx2nrbMw5jBRgNBBfHyCbjh2sg

b3stLxGmWp1CUlwuSrrbfhqyu2c3wrIMeVaWtxnONRB8zEG1NO7VriYNVgS64u4uPFu1wtF

zXFDifzNVBBJPRxSDTKwXi0m2kySEXGq+zf/wCrdzliLEKnE58Yxd0zqZzfmAJryOksQ2+vc

gf7gHm2k0+A49U4oWYLT+O5t7tmBc0XFgSG7j03XbOvHBg7n7NFbnrFcPxGGWLD4Jm0R

0xNEbAGgdgbb/myhVeZsGxwTmT5+jlebmGSxZfvz2Wz4xTVVE+FuK4RVUzy0smcWaGzHf

fgW7ea1GtwnDcMlfUT+HiFM8FpY46Xxk8bjlRtYEuShmdX0p001RqjvdrXcKNNibg4GrjJcElR

/gpHSUMpdC7iN5uQoUuJ+K0tljHkiUM8kqWESZZ4aoeJSyGJ/OgnZyxtZHVR/Lz7OGzSTwq

kyt/yO9kIveXG4IN0LgvQk+C5hkqKXU22qI7EEXFvVHw9rmVXjYW9kcjtnRuP0PHkqukxOS

EiOobqbf8AWOQtpwzD4q2aM0gYNdtwbD/hRS+lchxWTcspB8rXU7CYZif4sLnhrmfYg3HqF

PhqaTJuPUuLVTJ5gHEkwSNbJEOxuQQd+1lFxbBsRZhDBWUNMBTs/hyyB8cjfZzbtcPuFpc

GLudBNA+T+Lcad733Vdx3BL6Xk+k/ww9ccPxmGHKmLOkhfYuop5dIbNEeASDYOF7W8gv

Tx0iwaQe+xXzB+GKGsxGaoiwycHFKaNtXTsJ2e2N13NHcO/TYjtcd19Jcq4g3FcAocVg/RU

QNkIve1xcri/LaeNNm6Ps3tJa5wwy3duLeSY4tA37p2xFxvdMLb8rJRdTAvG/CE70KM8WBO

5UeQ/gphMC/cbG6jyE8Ir3XHCBIe53TP8jg3ILnWNiCnvJvYbILhc33QNiN8cQBugkg9k57r8J

huuncjIwI47bJl78pDykJsgYQ17r8JuoAbpSd0N5v2QNjpGFxJ3ulF00eVk77HdAOkZe/F04C3

KaB3snC/dCPgcBbhObtzwmi1906/bsmYSQ8DdOTWpwtwhbwEhlVtTvcSRYG9hc29PVeD+

omIyUeaax/gCMsxJzmxS2Be3UX2JPdwAt33t3XvKeTw4ZHXIIaT38vNfOrrFiEdf1MrYB4r2N

xaAu1XDi5jCD67kd/ILb8C8Xv+Ch5FZqX8lZFhU1Vh1dmaGZjsTrI2TPiBGmJzgBGywFhpaX

fe5K5uenFe+Uvw/EpGGQOlnFO97Y4m3/U9wNzfsPZdEwjEqjL81XQStIhqKh7Io2kXs1trut3

Ow8gtMq8zVz8Sx2ljlmZHiVb8rE1hJcWQnSC33Ok7bnTyuwck0YX1PhFPmGnxDLdJWYDX

YtVYthTrCN9NNbQ+3JBBuedr2XIK8zRSviErnC9hrbpJ9wtz/u3XYdi039n4pVvcJXl8rJHMud

wSO5BH7KXNk2qxSRpr2a3gWDw2x+9uVTu1EIdl7T6SU+jls8Uz2k2KrJ4ZAd2ldZren5iFo3n

2VPNk57DYg/hCtfWlhMty8fa+MHOBBI/doO3KxrZgdwfwuhtyi1vY/ZqKMl6t2i3qQgeugx14y5

9I0elj3+qPVfm4W0Zfa6GcGlcIHO7OaXNur2jyRI6Roda3oN10rK/S2CaJksgcNRAv391Uu11a

4yWIeMufo1qmwvEqukIM3ive0/9ObYH2PC59j2A1GCV7PGc3VJd1mg7brt2P5PdhXzUzQyz

YiGuHmByuHTtcXSmqqnyPa8gNcSSB90emtVvKKmo07o4kdU6JZkrso5iw3H6Ak/KVIEjB/3I

XCz2H3aSP3X1P6e0smG5cp8Pe2zGatABuBve37/zXy8+GTAY815wjwiqLTAx7ZngusS0EX

H4X1fwuBsNDFE1tgwC34XNefwpqPsv+PTw36D/AKbbJrndx90RwBQjYcc91zuMGkgEhJKjv

upUjRa/mo7xsgkxdEdxBKA/khGkAugvCaLYXYB5sbITm6jdGeB2QXE34unEbqbnumOd2Cc

9xA2GyGeLrpWZQia52yVx3Te26BsQJx33TfdKblJueUAcTAQOUu5O3CS9jsnAkboQsDxc9r

LAN7kpAduVhdfa2ybgLA657JWnUUwEA7pb2OyBjhgU5t7oYJI3RAQFHkWDKlpfTyMaLuc0

gD17L5z9bsHrMJ6jYm1jyZYcUZM9zewLC632Gy+jRLnMLWu0kjnyXiH4hqWKPqfiETWhwq

Q1jnWsbiItLvUm5K2/Byxe4/lFLXf+Wf3OGZlZVnDa+spJWtc2oLfEH6z4oYCP6/8AyKBlLF6h

1XTQTUMVRNSRaIJGgh8bQC25I5/zW9yT2WyVWCyVFNWYM513z1ETo5OxFhc/hg/K6D0

Y6RSR0M2KVtEQZn97/SwbBv8Ayuous+OBnaWh3TwjX6PIdPXU8eqENkeNX0t2H5TP7ixU

MpbMxzrb3su94lRfKQsjii06AAB7LWMdqKCZt5gwP7rDttcjsNJpY1rJxbFcmNlvJBFt5LWanK

E7HEOp3Wv5LtUhpZGlkbhyq2emi1/pBWXZbJM2q9LCayciblFxt/DP4U2LJspZtH+QujStgjIF

hunsbD3CglqJInWliaLRZW+XeHSU97eQut3wiCKKIM1AW4FrJxZCXei2DDG0hpxct1A7E9l

FG1ykQXU/HE1POWDxzYHUTRx/UYi4n1XkXGKN9HWyslZuHGx9Lr3zNhEWOxupmuuDH

ZwPG/ZeV+uXTaryziLZ42uNPLctP33C2/H3qtuL9nMeU07tipR9Fl8Jcdazqrg7aR4idNOQdYO

l7A03affhfVbDBppI26Ht+kfS4kkfcr5g/Cxhb5uqmDaGveyn1OJBsW7WBB9CQfsvp/hhmNDE

ZyC+25ta6yPOS3XJlbQx2xwSJN+NtkAncnujOCE4b8rAlLJoApLKPJ5ealPOyjPF9yhXImR3jffso7x5KVIO/BUZ4Sk8dCQB/khFpJ2sjFpv7Jhbc8IVljm2EjumOfyAmk3NvNISAumkzKMvbkp

heSCAsL73TTYIGx0siE9ymlx7rHG+yaBugyGlgcNzylDu1029ll7pmEO1EGwN/NO1DyTA6yS

90GQgl7p7TsChApw/ZM2IMHbJ7fwhtFkRvKAQQbiy8c/FJCzDc2vxh09PpLR9Ebw57HBouC

L7XudyBfdex2DdeEuuWH49m/DcSxhsBDoMQqKGSVkZax8kRNgO19D2E27n1Wl4q1U3qb

C/Qz11UlD0adkeRubMw5ZoqR+v56qGk32s1wBb7ixC9yT5Rw3KOWYqeOWFhiYXzPc4FgI

A2v5brx103ynRY1kWjd0xydW0/VvCom1VIYakindGx+iWVwleIg5zd7EDci1yVq+HdUvi16nZh

rOm9HXVP9oUOsV0NXQxxCkDbh3ilzLs32ta5NgASux+SuSy+cmHCNlLcemmbZ1l65vjmm

wrLVO+OOMlr5pNnvPew7D9/ZedMQ62YxTzOjdUyE333uusZr6dZvwYRUmdOonzuITN1Op

aOjgjEXmXSPGzQe5DVx3G8sYUxz53moqIiSBKdFnOHO4YAfsqsaqova0WbdXe0nkJhnXz

FqeoInmLoyeCFveE9a6DEKYGoGmQ/wCncLis+B4VI8ltM8NHJ7/yVll/KBrpmf2RNqe038Gd

3h6/QO4B97D1VXUaWmXouaDyupreHLKO84PmynxeodKy7o2C5JUrGM10tG0Pa7ZarkuX

KjKCpo8QxtmD1tPIYqimr3CCVjxzs7la71ExDAaOjBw3MtJXSOdYNhlDiB62KyP0ylPbg6n9ft

p3to2HF+reG4awl0l3eQWm1nxD1hf4VA4Rji9rlczr6CqryJ6t7tLjdrb22UaLDcMgdYO8Mna2m

+60K/HUpZayc7qvM6ib2w4R3TJvxA5hocQjqzUiWNxAew7hw8iP9rLq3UnG8E6jdK67HaKL

TVUYa8xn9TCSAfcbndeech5Iw7GK+Cnq5HNMrg2znabFwu24uCNXY912jLfSLJT82DLmO

YLnKpoqnCnTmPL3jzT+KHWBAuW6bX/ULXIUGphVBrHDQFN18ovdymbn8DOCQyZyxdtX

TD5ulpLFj22cz6gDzwQSF72bGI42s8gvlb1I6PdT+jzf77ZYqs24Rl+tqBTQPxCoFPWi4u0Sthe

bcHm3HC97fDb0y6h9NMoVNN1Hzq/MGIYjKyoDXSyyilGmxYHyG7u17AC478rF8nWrP7+7/

QqJOL2NcnWXDyQXEHzKO5AdssF9lwGTZBfa9yiuPPkhP5sn9CASb8ILgpDgSUJ3JTL9wk

RnoW/sjO3QjsbJ0xGyEgIbnEiyQny7pbGwuukkzJwM43SF3CV/Nk0qFrAS4Ev+Vh2CzyukJ2

2KENcmFL6d0gS7+aHI4vCwLAnbAJDozZPG/CYB2Th7IGEGbxuntQ2hFaChEEabWK8f9T

Mx0eB55qsnVELhhzZpA86thJO7W6S3o4t+zAvYDQvI3xF5TLeo7MUbH/h6mSFtQ8D9Gqw

BP2H7KxRLnabPhXH52rOsM3b4UsrmLGcz5kkjBZAxmHwuadgdRc8D8N/KFlmi/u31U6vTw

Ma2fFqzCq8ScudC6GYC9/KVs9v+F0H4csOjw/pjDVMhMTq+WWpcHfqtqIF/WwC59n58+W

utmH1FW9zaLNmCzYc0hpIFZSvdPGHHgXjkn0+diu0z8enSRzk61qNTJI5fnHA8uYpj8tRmX

DnVVPNqL9EhY+9iAbjyNj5bLk/UzEcZxmiGEYjiRrmQSB8NQYWiXYWs63OwG/K7HmsN+Y

ke83sSLfdaTVU5q3aGU4cT3twsj9a4y5Nn+jxthk8/R4BVyRvp2QOPiEanObZbxkfpxVRhtVP

qYz/LcWJ9V0zDMo0VPUNmqmtc8kEDyW3VeF3o44KGMeLMRHH5aibD7bpT1btaivY1fiI6S

DnPnB5y68ZIOO9UMu4NlunHzmJ5co56wi9jI0yRh58v4ccfvz3WhdRei+Yun1FT4vWTR1FI+

Rscj473jceL3HHqvSXTbCaLMOeszdTWMlfQTzx4bgr5QQX0dNG2FkgB3s4Rt/BW158yxT5

wyrimXp2NvW072RucP0ycsd9nAFQXeTlRcqo/auyzovBx1Old0/ufKPLGaclVdBisz6RrjDHM7

wiBcWB2CopMv1VZXiadjo9bgXtDNiV6TyFBFmTp/ST4rhrI6umDqWrjO+ieImOT2u5pdbsHB

QqjJFMZTPhrhsfqjcf5InrHW2iCPiVNKRGyhkrC82UmH1uNtLpqERwU4i+jRGzi55K9W9IsFip

83NrYANEeEywOJ53ljI//ACXn/KzJqCVscjdO4uLcL0P0nqx/bEeg38Smew78bg/0WXrL3ZFss

3aX4YMf8SmG0uP5SwTK1SARjmZMNoAO9ny2cfs267ef0gAbLjmM6M6ddct5ca3x6LKVNLj

tZ9N2NqXAxQMd2DgXax3+j3XZXC6yZNqmFb/d/wDTNS+tsjuKC8W47qQ7yCC8HyVeS9ku

QDuEM+SI7jlDKBDYButwVHcbcBSJOCQoz00mEBeSh6fVGIumEb8JkOi7PPoElz/qKVwvvd

YAOP2XStmV0NPoUltk7UDsEw37qN8C65Mcm8pxSfso2yRcmHba6UBYEvHuhyEjLHust5r

BynAbpsiwYBxcJ7RwUlk9vFyELHHBEBTBYpwG6BiCtK5p1ny3DiGEy17WDxHxhjj/AKtBu0f

uV0prd73UbFMLpsYo30VU06XbtI5a4cFFTNV2KUuiam11SUkaD0WxMMyPQUrdTTTtexw8j

rN/5rUOuWBTZro2RUdWKWtpKptXh9UG6vl6iMfS63cWLmuB5a8hX+VMPrMpYliuB1b3StZ

UPlieBYGN1zf/APfVBzGXVcbnH9LCTfzXXq/5KcIbT1J6nf6PPNfmikhJpc40kmB1rSGPlmaX

Ucz/APVHOAWgG1wJNLrchRGYxgLwTh2N4ZUesNbE8fs5dAxunD2eP4Dmaxq0u5A8iufYvhlC1zhJQwOdzvE0/wBFkWuKeGdVVGe3EWQJsz5eo5i+tzHhcbmblhrIy/8A+kOufsFPx7G8

UzxhTMrZIw+pihr2eBWY1UQuiZHTvbZ4p2us5znNLm63NDbElt9iKKTCh4T5Kaijjc0EtLYwN

11KhzzkbL2R8MmqntiqBCxtRI91iZLWcD909U4r7exr6XZ9M+i7psmYXl3L1FhuFxaIaSBkTGj

caQLKp+WAeAfOysKzqvlTFcvMGBNi1NFnyNl1k8duy5zV52jbUttUAb33Kz9VVifHZe0Vq+P

Enwuiuxqhr8nYzieK4fTSzUmITNdiFEHhoedIAmhJFmygAAg/S8WvYgOVfBmLAHBk8uKS0

Tn7mKrw+pY9u/Bc1joz/wDF5W84jmzL2O4fUU7vDfUy0/huaw3Oq1gVorMMxLD3BpbrjJuLfy

TxtzHE1yiJ1re3B9lvQZky62bxIq+prHDcspMMrJz/APbEVuWWM/ZspZ5XZQylWRVD2OiircX

aKaCO9vq8O5ldtwCwetlVZbGsB5vwLg8hblSQPfVQNj3LxpPuorLYYxtyV7dM2nvfB1noBl2T

BsExfEq2qfXYjiuIGora1/6p5AwCwH+Vrdw0dh5m5XUHH9lVZPwtuC4BTURYA8N1SW4Lzy

VauPYLOtm5yyc5JJSe3oE4eiDLvtwjuKE4cqGQwBw2PdBcOyO8eSGQeVFn0P0Ac3bSEJz

D3UhwCG+3mljI5FNjtv6Jm3dFfa52Qjyl0OXN+yafO/ukuTynDcbrpGzL7G7g78LLJxta6aQUD

4DE2WLO6UBA1kXRlrcrAEpJPN1gCBhJihLue/2Th5BKBuhFkxjfNL39ktkoseEOR0OFgnAJ

qcDtZM2uglyEF+6ewC90xp809m5uo2x0jRuoNMafFaXEQHaZYTESHWF2m4+/1FaTXVRe

zS79LtgQdreqv+snUDLmCYnguUKur/8AUcTm1NjFrRM3s51/M7Ae599Oq6uF7y2Ka4BAG2

kWW/o5yVC3FnTYcmkUOYoxYBo1F2w8ytarMuMjJmrD9ThctB7LbsRkhY9jiNTWbu8wuTdV

c/zYdQzuoS6N42aXAAAJ41fPI15av9NDLLZs2Fyh7WjS1twSey5N1amy98o1rWNkdqJJY+3

3Whf39xmrqfBiq3HUdx6qfNlKsxan/tHFsYigjdvZzt/sFPDSKqW5sqz8k9XBwguTn8mY8VwnX

NRVb42h21th/wAqFiueM1VM4hOIaA5oJcxoBst7xLpnQ4phmjDcbhfI030uBbf7qjpOl9ZPU+Li

9ZBSwRNDHO13LrK7GdC5lgy7NPq4rbHPJtXR3PuE4RO6ixqQudLuJOTq9br0LhuYMsYvp

o21LC9zQW3NivLByRhMLpH4HjbJqmMH+E+4J9iQq6gzJieGYlGx9S8PgeLC91nanTVamW

+DLmm1V+jWy1cHs2joYqCsaY3XZLwtzyxSmsxyipoxdxfcLjvTfNn94MPjDtpYwC7mx9blW2

ZOv1D0cxjCcXrKGOuhmnMc8NwH+D3c0nh17EegI7rE+Cfy/GlybFusjLTubfo9wxXELAebC+

yR26oMh58yx1HyxR5ryniTKzD6tl2uAs5ju7Ht5a4dx99wQVfkrMmnGTTOfXK4GO3Q3egRH

kXsEJ5/kom/QaBuFtrIb+bnyRHboThZM1joSBP9EF/nZGdZCcbpxwD+EInfkIkm52Qzyo5MdI

uLDk72SFx7JSVhuQulZmIw3Ius55SC/mlso2GhOAsWeawBBnA7QqXbssG6d32CbsSRgF0

8WtskA25Si3CFhYFA8kqS/ks3UeB+hw8rpzU3ss1E8IGhZCi3KpM75xw3IuWK7MuJPHh0kZ

Mcd95ZCPpYPc/gXPZco+IX4lKDpA2ny5l+lhxTNVcWeFTSAmGna4/S6WxBuezQeNzYWv5

N6k9Yc85wEdJm3MJrpYbuLGMbHCxx5DGNAFhxc7kDcrV0HirdW1NrESrqNXClNLs0rrT1V

xnG81TZnqawnEaipFQCwkCOxGkNvwBYADsAF6PypnijzdlfDMyUkzDJWUzDK3xf0SW+ptv

Q3XhHqFXvkrvE1HhXHRLqliWVcchwP6pqPEJ7GO9tLyLXH7Lq79EnWtnopeP1sqbHKXs9

0x42+SF8Mr/qHnwF586uvxrMOMS4Xh0f8BrxaS+zjp/pddWwnF4cRkdE1z4p47seyRtrEctIP

dalm6jxWhqvnaelvHIbCRrdQH+yya24s6G6XzxRz7K3QDGZdNfPjDoXu3Fjey32HozQtgH9q

YrUSygbPbJ/QqVRZsq6SnaGRPY8DSdtitOzNnfHKOWSWEyC5J5KXyTnLC7L+mqpoipNFli

PSwQtLaLHn7cCVoNvwquHpe+Z4biGPCx2+hpv+65/ivWvHaRxZuSNlFw3rTmCrk0inBJ3H0

oJ6bUNZLy8p49va+zslP0Vy22ldMMRqXz9nGSwb67LlWdukkmCTTYjQV0kwjN/rJOpXlBnz

MWJNDZzIGki7Wi2yv8AFMTq8VjjpALRsA1EjlVZSnU8FTVyouX0rgd0uxiowPLbg9lpJBYOI4

C4F8QPUA49maOnpJA6Kjbo1NNwTfdb11Qz9HlXAZcKw3S2pqGFmof5QeSvO9N4uJR1D6

l19W4PclaXjaH8nzzOb8jqNsPhies/gk+I3EemmaYMGxSvc7LmLSNir4n7iInZszfIt5Pm245tb6

d47nPK2W4zJjePUVLpLWlj5QX3PH0j6v2XwryHVvpJidemxv8Ahdioep+J5lxWuxmqmLairqX

yPDXOIBJubFxJ58yVB5TxMdRZ8kePyVtHrNkdkz7Bw1EFZAyqppmSwzND45GOu1zSNiC

OQlc3zXmv4L+puL5py/ieUscrWzuwjw5qNznXk8N7na2m53AIBHlqPovSRcD5rjdTS9NY62b

NclZFSQh22KC873RHP7BBN1WzlkmMA3E9ghm5RHWQZHjzTt4EkClB5FkIl1+QnvJKGdX

+VRvLJEi4Sg2HqhBxvyngnvZdQZOBfusFwsFrg3SqNoIzlYfRYBcXScKNjocPNLqsmLL9gmyOFbuEoPcoYcG+qUO4QPISCE+iy90NzlEq8UoMODXVtdT04fs0yyNYD+VHl5wOT737p2q

wXEerfxQZQ6bRuo6CI4xiV7CNj9MLfd+5P2FvVeas4fFj1PzMXVNDjQwmnH6aegBYGi3c3L

nfc+1loUeN1GoW6Kwv3K1mrqreGwHXuSpj694rU4i8yvjNS+NziDYCzWW9mXC875jxyo/tt7

3PNtfHmt7zBmWux7CaTNVTL4tZRVD4KogbujO9/uCfu1cwza4OqfmYbOY/6muHcLtdLB1U

xrfaWDCtanNzKnNx+aaJQ0bjYrW8lVBhzjhriTaOoa6/sVd1k5qqEtJu9q1zCHRw43DODpcx1r

8bqzKL2tMeuSU1/J9GsUy6zNeBUWZ8BnbTYoYI3P7MqG2Gz/8Ay8nfY7cUGH5poTUPw3

GY3x1LHBksL2kaSe/sfNL0TzUzEMpQUck2p0LQwj0Q+pGVKPHKhtUx7oapjbNmYbfY+a5

WqxRk6rOvR3E6Pkirazb2UOCSMbIynhLbbEtvYLWcx4DlicO+YgjcXcabbLl9TjWacrw+FiTX1

EDbtZLGSdvUcrWavqjIJHBztO/J2Typk+YjLUKP0zRseYOmOUKuQuEI16r7GyFgvTzK9FOB

JTsAvtYrS6vqOJyXCtbceqhSdShGRepBI73R7LnHDY3yaZSzg9DYZlvJ9PS+IyBviM7uK0Xq

Xj2DYC0xYcQ6d4tpaODZc7h6mYpU0xjodT3m/wBVrAfdUdXNU1Uj6munMkrjdx7KrDSfXmT

Gu1MZx2wRzbO09RX4lUTVMrnPcSd+yoMOa+OCUgXAF73V5nWeJtWWA/WeVQSztpcOd

e+t38l0FKagkjltVNOeA+W6hlPHLO8gEE2XQMg0RMbZH3Ac4uv57rmWXaObEZGRRghpd

9R9F1+hecHw4OiG4Fmiya5NvCK8Gl2bvged8SylmSjq8tV8tJX0sjZGSxOsWkf08xwRyvbnR

v4qsJxyiOHdTcYoMOxBrWCCdsEjRUc6i/SCxp4/0jfYL574bCMPlfi+JP0ySDVv2CFS9S8NFe

6TwKt4voDmxHTYLG1vj4atYl/0v0ap1dH2FwjMWB5gpTWYDjNFiEIOlz6aZsgabXsS07H0Kl

uIPJK+VeXurQwyZtZhNXWUc4Gz2SOjcPuNwu45a+NDPOHxxsxD5bFGNaB/HiGoj/3M0kn

1KwrfBXVv+28mjHX1tZke3n77oZZvfyXAsjfGDk7H5W0uaaKTCZH8TRAyRA/+Q/UPtddvwv

HsHx2nFXguK0tdCQCH08rZB+x2WVqNLdRLFkcFyu2FqzB5Jbu/ZCOnuiOcDdBPKrYJGW

Wrsn6iUDWsEl10zyjMSJGo23KdrBUcO3TgfNA1kcLqCzUOUPUsv2UbCx+Amq/KW45KHda

5njqBljp7hJxfNGJNpoiS2KNo1SzO/wBLG9z+w7kJkm3tSGbS5ZJztnfAMgZfmzJmSqdDSQk

MAa3U+R54Y0dyd+SBtuQuHw/GHQxPqDimXKenp4nHTM6u0NLewsWm59lxn4m+sdR1Hd

hGE0kUlFhzv8QyJzruDBu6R9tr2bt72XnioqYcUq3SzsDmtN2+IdRt910Gi8VCVe69cmZdrJK

WK3werc9/HdQSQSUeW2yM1NLS6lhL3/aR1re4APquA5j+JrMuL1Tp4aOrLrWE1QHSyH7la

TVTse4sjADQo1S4R07iAN+62KdFp6FiESlZdZb9zImN9R6vHKt9Ti00xc43u5hFkAZtdBCY6

CB9Q97dyRZoVc6TXKddtN09hMz/AA2Cw42VtRSRA1nk2nJeJ4z4s1LjkMDMNxRhjFnXLZ

R+kn03I+6psxYfVYLM+kqmF1LqOg92/wDC2KsofEyu1jCdbQSCOyosNzdFigbg2Phpq2/Qxz

uJB2+6BNt5Q6fo0qqqG0jy5rzodwqirY3WKmMEEnUCt7zNl3DaVuusZNSeKNTNbCLjsRstO

qoqCRnyzMRaQ39JOysqaeMD1rnk770Kzw+kiZHJIQ0gNcu7y47883W8AtPBXizIWOQ5cqv

CrqpojJuHtBIHvZegMudRMCroBTU+KQvfbZt7H8Fc3rNPKFrkkdl4vWRdShJm+4m2kq6Yh1i

bHlcpzJlzDnySPbCw6uVdYnmJ7XO0S7WPBWqYhjch1O1cqvWpJl+512FBPlnDwS11O0C6

GMAwaFwtTMcQi1FdNUPvcgIlO2STYtuVLJyxhFTbX0Z4ULG6IYmsHk0WVbiL20sD5XcAE

7rYG0Tg3XJZrRuSSue51zDHOXYbhp1gXa9/Y+gT0QdssFbVThTDJoOJzvxHEpamXjUT7D

sqzEfFrZWUtKwuc42srU4fVuc5gkY3V3ceFdZewrDKQuMNXBVVYGoxh4DgPQHlbanGC/g5

hp2ybZPydg7cLiY2Ruru4rb56mhhPiyygtZu1nqtPdjtQ8eDRRBrnc37KuxeoqYqRwdIXyP5Pk

qyzKXPsdpJceiRmfOM2L1ow2id9F/4juwHkrXBvDdG0NAJAsufUjmxzbg7u3K37LUf0eJvblS

21xjHCK0JylPLLKuqZoIiWuALduFAhxicESjUHja7TZZishe8xtuB3UGIaWlvqquM8FhG+4Rm

qINaKpxbt+oDhX8mecfwBsWJYRik8BafomgkLTv6jcLlLbuBG9uxurvLtWZjNg9U4uhqWFtj2P

YhRyqTX1LJPGUmvpPY3Qn4tMSpH0+B9SK51Xhr7MZiDwTNTk8ayN3tv57jzNrL2FTVdN

WU8VVSzMmhlYHxyMcHNe07ggjYghfIjCaqRuHVFLJ/1YCWEjzaeV0/p58UvUrp9lxmWcMx

OGakhkc+EVUXimJpA+hpJ2bcE28yVz/kPCqclZp+H7RoabyDgttx9MGzJ7ZAeVCEnkiNkuoc

k5MEieH7XKhiQ9k8SE9ymY/ZK8QDe6USAnlRQ4nunA2tvuoWxIfW19Ph9FPiFW/RDSxPml

d5MaLuP4C+bPVPqbjXUvP81fXVUj2eKYqeLUdMUYcdLGjsBf8AqvZXxKdQYcn9PMTw6CX/AB2JUr4wA7eOI7OcR63LR9/JfPTK9Y12MSYjOSW07HzE+wP9bfldB4TTb91skZnkLsYrTL

TPuMtqMRqpo5CA1rcPhHbQy2p35AH5WjuqDf6XcKZmaqPzbIXG5ijAd/7nfU79yqLxTsBv91

0K+nhGXzLouIZw7ZxCSvl1QaQNlCgfsCT+6O92thCNyQ2GUFRN4UmmyssHAe8PJVRiVx

PvsrDCZRGGm6ZvK4GWc8nRI3NdhUcfN73XJ87U3y1UKmK7JGO1NcOQV0WkrQ+BrQdg

tdzXhUeIxFx5PKan6JZY8nlcGuYhimJVOVKHFa3EKmr+ZlljkZNIXhrmkbtvxsR+FSy02GSNE

rtJuL7Gyt6ukNHkqKBxJ8PEJbX8tLFqr6WknP8AiJJGNHBarMZRfQ2WJJA2N2ukqSLf5Sbo

DscrKd1mu3Z5p1VHhAsIapzbdxv+yDDTUIJ117HB3m1KSjjLFGySfBaYbnnHKKZsjKmYNG

5GslpHqCtll6mT1lJ8yyGNhjtraeT7LRpniijIY1sjDsHDcBWGEYTiGNUBdRxR6IJnOfqdYnbhV

ZV1z5aL9WrthwmWZzy6uOnxXN1HaxTv711UbhoxKZvlZ5WnyUzoZnxuBYWOP2UaV74H6

m3dfi6ljpq0RWam7tM3l+Zqup/6uIzSNItZzybKBLXNuS534KooIcRey5DWl42B5RIsGxKd7hrt

p3Nii2VwXBXdtlv3PJIrqx8MLpWSXc7i5UukwyBmBNxdsbnTmQHWDuAquLL9ZV1Ipw8uAN

3HyC6PguGQxYR8kBdjBwVFbtS4EtzfBz6fMFQ+draeJzHE2cT3WyQ05qaIF5Jda5PqoGL4

S2CrJiZyVd0X8GlY0243CitnFQW1BV5be41qagdBONiLlb7gEfh0VzzZa9URNqJ227lX9KXU

9Lo7FVbLHJImUYx6IldK0SkAcqIH6jt2Tqk6pHH1UcPsb/umSFn0SmS6fpBsnUVc6nroZm/6x

dRdTSbBDcS0hw7G6ki0+A9203E1TYKqoc02bP8AWNvMKonlIkOk7HdNfW+NQxTX3A0OU

JtVtuQE2It8hSzLo+zzHXHKMHW2uocctrBFEmxK5A20iW1/ZEDhyojX3tZFDvVA0OHD91kk

7IY3SyPDWRguc48ADkoQdc8rmXXvPQyplF9DBNoqcSa5hIdYtiH6vzcD21J66pWzUF7AnN

Qi5M8xfEpnp2Z67GKiKdzoS/w4Qbi0TSA3btsLn1JXCss0urDZXvcWmunZTNP/AIj6nfsArzO

OISYuyeGMlz5nta0X3JJTaimiw2opsLiH04bQvkdbvK4hpP7ldtTWqa1CJzVs3bZuZoGOzOm

xCpmJvrlcf3Vbq73VhiTPqc49zdVpsTYqxGGewYMkQzE7KdE/6CSVWx2FvRTY3C2xTSi+g

8eyrxOFxfrA2WYe8aQCVYTx62m+6rYR4cpb3uU2dqE48l7SVhj+gutf1Uya88V9VyqEvtwT91

YUVSXsA1b8EIovKAlHDyVmbmGmyzDGAPrqXyW+w/2XP5XbaxfbnZdBzyQ+Ckowd2gvI9y

tMrsPfRBkkrD4b+TbYJJ4YslNPTYbUtu+R9PKTza7T/sguwqljfZ9e07XFlLqqOUXdDH4sfcDk

KIxsLneA8OFt9+QpIyysZI3+QxdSUtK6Jk3iOJG19luvTa89FWA8axt9lpLsPjk+qEm49FsuT8

wUmXaaaGrhlL5HX+kC1k9leYcCjZ9XJUY/SmHF6lo4Lyf3UKmpiybxHgOb2aRe5VjjddFieIu

qadha1/Y8o+G0pA1NZqeR37eyGUpRjtJI88mU8JLi57gHO3seymtcXAU1OOT9TrcqVQ4Ow

u8SslsPJWrKamaWRwMFh3CglEJMh4fhroGue0/q59Vf4ULNcx3cIb4w0Bg4CdTXZIDwFE8r

klgkygx2EvnItwbKPHI9zAwngWV1jFNqlLgLXUKjgYA76fyo3LKCcMPICmpnyzNdwAVb1BDY

9IOwCbCwDssq9oyRvsgTbeBbSqkO6jOdpRpnX+kKI8m/wCyU20KMcDy+9vZI6UAck7oTb3

7pridV78FKvsT4J9K/VRTt3Gg6h6KKHuPPbZEoXanSwj/ALjTshW0/TbjZTYx0Dn8n2jY7yKM

1/ZRWbFHauNfR0KeCUx3knteb87KOwnhEamXI7YcOs25NrLw38RXUF2ZM11r6eVppYnG

GCx2LG7A/fc/cr1/1ArqjD8jY7WUzgJIqCbST2Jba/7r5wZyq55KmV733JK2/C0Kybm/Rk+Rvc

EoL2U9PiZhqzUOOp7Tqb6EcFEOL6/m62V/1TGOBt+4AJP7kLXHyP8AEJv3UTFqiUfLMDrA

XNvW66mVaZixeeQ+JvubX2uq02F/NS5iXtu7sFDk+k7IIomj0OaSjsG9j5qM1EjcbeyUmFF8k

wt1Mt+FV1cVnk8Eb3VnESWXPfdQ6ndpJVef4JmuAVNL4gIduRtdED3xyN0EjcC6hw/S9xB7

qZG3W+MOJ3cP5o1wslZvLwDzK7xcRaxxuWRsbt7LHwx1lGIZGg7WUbGSXYxUA9pCPwi

YfI4EAFBMFP0atUwTYZVGPSdN/pSOio60/wCJiBdbZw2K2fH6WGWnErm/UB2WuwNaYyS

OCkpYWV2HGPoHDhzIrmF7h6IktPC6LSYwXdyURkrwG2sL7FO/Ve6sVybjuAa5wQoqKMP

DrXsrSgux9gLC+yjfpc0DvZTIgG2ACVksksVgmF2s3JO/mrfCIKYROmkf/E7D0VOze/op0LiNI

Cgz7Ayslm4anGywAt3KHTuJBBRr/wAO6gnyyet4IOLHWxpA90CkhtHuVLrGgs9kKLZlwoZZ

wTKWXyFEYbY2UaseAwhSQ4ll1WVl9drp63uCm9i4K6XnccqOSeylSbsLvJB0NMeq25Knlja

V+W+ARNiCmENJ+o2T5xpjBHmE0fU0EoINZDl0Epm6KiOcXsOybiEroKp7GtBB+ofdLASX

Oan4hG18zHOvfwwj3J9kU0/R/9k="}

}

\*Sample outout

[

"{"identity":{"0":"images/Ajay.jpg"},"source\_x":{"0":113},"source\_y":{"0":109},"source\_w":

{"0":135},"source\_h":{"0":135},"Facenet\_cosine":{"0":0.1823376272}}"

]

**1. getServices**

This API endpoint retrieves the list of services that the end-user can avail.

*Request type:* **GET**

*Output type:* **JSON Array**

*Sample request:* http://localhost:3001/attendanceLogs?attendance\_id=100046

*sample output:*

[

[

{

"attendance\_log\_id": 10000010,

"attendance\_id": 100046,

"user\_name": "ajay",

"room\_name": "testing22"

}

],

{

"fieldCount": 0,

"affectedRows": 0,

"insertId": 0,

"info": "",

"serverStatus": 34,

"warningStatus": 0

}

]