



# สมาชิกในกลุ่ม

1.นาย ปวิช สังข์วารี

รหัสนิสิต 6320502452

2.นาย นเรศ เพื่องเวโรจน์สกุล

รหัสนิสิต 6320503041

3.นาย ภัทรพล แจ่มจำรัส

รหัสนิสิต 6320502479

4.นาย โรจนากร แย้มบางยาง

รหัสนิสิต 6320502495

5.นาย ชนกานต์ ศรีศรุตีพร

รหัสนิสิต 6320502371

\_\_\_\_\_ data warehouse and data mining \_\_\_\_\_

# Loan Approval Prediction

\_\_\_\_\_ data warehouse and data mining \_\_\_\_\_



# วัตถุประสงค์

1

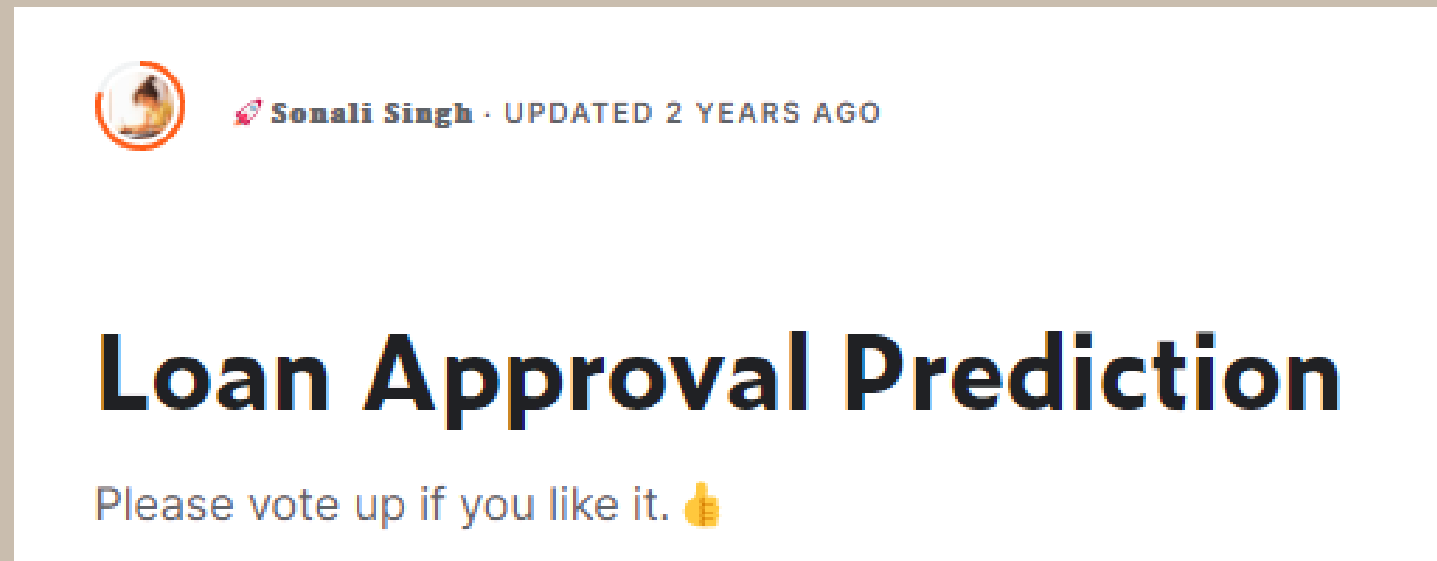
ปัจจุบันมีผู้คนมากมายขอยืมสินเชื่อเพื่อกู้เงินจากธนาคาร ซึ่งพนักงานธนาคารต้องตรวจสอบข้อมูล ของลูกค้าแต่ละคน เพื่อใช้สำหรับประกอบการตัดสินใจการอนุมัติสินเชื่อ ซึ่งใช้เวลาตรวจสอบและประเมินข้อมูลทั้งหมดของลูกค้าแต่ละคนเป็นเวลานานอาจทำให้เกิดความล่าช้าและผิดพลาดในกระบวนการ ทางทีมจึงเล็งเห็นถึงความสำคัญในกระบวนการตรวจสอบข้อมูลลูกค้าในการอนุมัติสินเชื่อ

ทางทีมจึงพัฒนาเว็บแอปพลิเคชันในการช่วยตัดสินใจในการอนุมัติสินเชื่อ



# Dataset

2



บริษัท Dream Housing Finance จัดการเรื่อง  
สินเชื่อบ้านทั้งหมดมีอยู่ทั่วทุกพื้นที่ในเมืองทั้ง  
เมือง และชนบท

## Dataset Description:

Variable	Description
Loan_ID	Unique Loan ID
Gender	Male/ Female
Married	Applicant married (Y/N)
Dependents	Number of dependents
Education	Applicant Education (Graduate/ Under Graduate)
Self_Employed	Self employed (Y/N)
ApplicantIncome	Applicant income
CoapplicantIncome	Coapplicant income
LoanAmount	Loan amount in thousands
Loan_Amount_Term	Term of loan in months
Credit_History	credit history meets guidelines
Property_Area	Urban/ Semi Urban/ Rural
Loan_Status	Loan approved (Y/N)

614 entries total 13 columns

# Dataset

```
In [10]: loan_train.isnull().sum().sort_values(ascending=False)
```

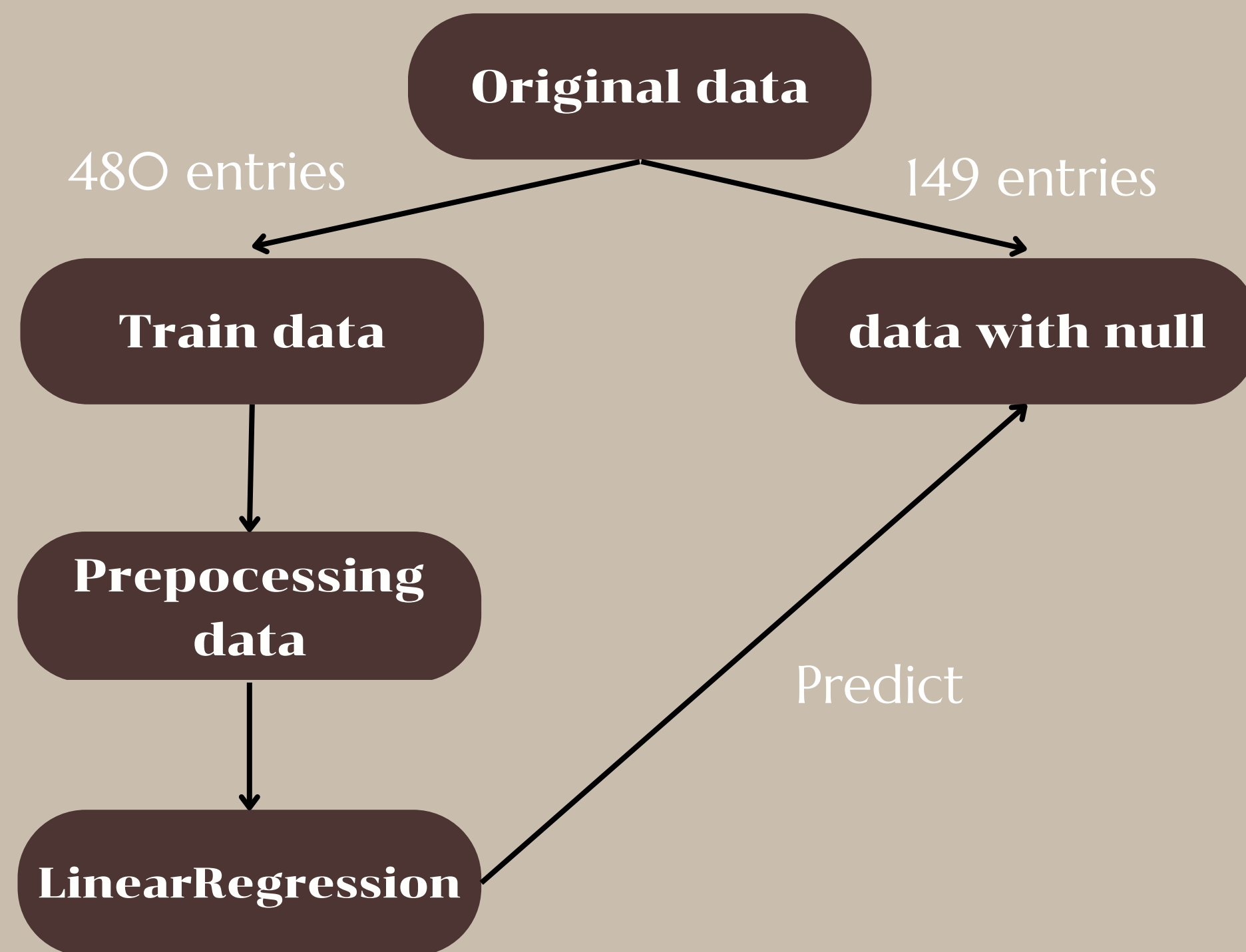
```
Out[10]: Credit_History      50  
Self_Employed      32  
LoanAmount      22  
Dependents      15  
Loan_Amount_Term      14  
Gender      13  
Married      3  
Loan_ID      0  
Education      0  
ApplicantIncome      0  
CoapplicantIncome      0  
Property_Area      0  
Loan_Status      0  
dtype: int64
```

Sum = 149 entries

## Solution

1. Regression to predict null value
2. Feature Engineering

# 1. Regression to predict null value



Y target

LoanAmount

Dependents

Loan\_Amount\_Term

Credit\_History

Gender

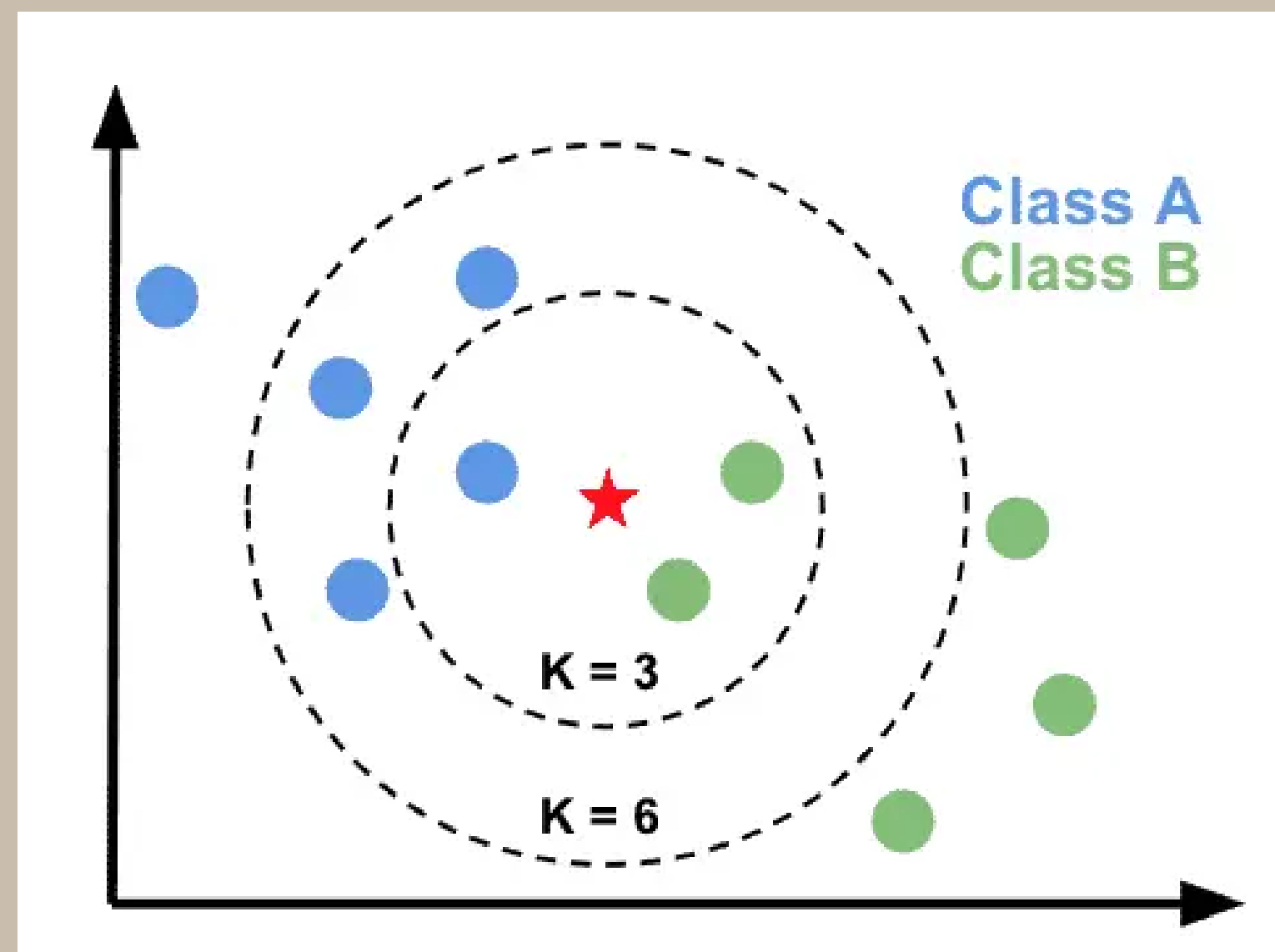
Self\_Employed

# Feature Engineering

## Gender

Original Dataset		After predicted	
Male	489	1	489
Female	112	0	125

Missing value: 13



ใช้วิธี KNeighborsClassifier ในการสร้างแบบทำนายเพศ (Gender Predictor) จากชุดข้อมูลสินเชื่อ

# Feature Engineering

## Married

```
In [29]: loan_train[loan_train['Married'] == 'Yes'].describe(include = 'all').iloc[:, :]
```

```
Out[29]:
```

Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_St
398	390	398	380	398.000000	398.000000	384.000000	389.000000	362.000000	398	
1	4	2	2	NaN	NaN	NaN	NaN	NaN	3	
Yes	0	Graduate	No	NaN	NaN	NaN	NaN	NaN	Semiurban	
398	174	309	326	NaN	NaN	NaN	NaN	NaN	151	
NaN	NaN	NaN	NaN	5629.173367	1794.632965	155.750000	336.987147	0.842541	NaN	
NaN	NaN	NaN	NaN	6989.254425	2039.962354	89.934267	66.739163	0.364737	NaN	
NaN	NaN	NaN	NaN	150.000000	0.000000	17.000000	12.000000	0.000000	NaN	
NaN	NaN	NaN	NaN	2817.500000	0.000000	108.000000	360.000000	1.000000	NaN	
NaN	NaN	NaN	NaN	3854.500000	1625.000000	132.000000	360.000000	1.000000	NaN	
NaN	NaN	NaN	NaN	5820.500000	2464.000000	178.500000	360.000000	1.000000	NaN	
NaN	NaN	NaN	NaN	81000.000000	20000.000000	700.000000	480.000000	1.000000	NaN	

จากข้อมูลผู้สมัครที่แต่งงานแล้ว 398 คน ในจำนวนนี้ 309 คนจบการศึกษาแล้ว 326 คน  
ไม่ได้ประกอบอาชีพอิสระ และ 151 คนเป็นชาวเมือง



# Feature Engineering

## Married

```
In [28]: samples_with_null_values_on_gender_column = loan_train[loan_train['Married'].isnull()]
samples_with_null_values_on_gender_column
```

```
Out[28]:
```

	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area
	1	NaN	NaN	Graduate	No	3816	754.0	160.0	360.0	1.0	Urban
	1	NaN	NaN	Graduate	No	4758	0.0	158.0	480.0	1.0	Semiurban
	0	NaN	NaN	Graduate	No	10047	0.0	NaN	240.0	1.0	Semiurban

จากนี้ เราสรุปได้ว่าตัวอย่างที่ขาดหายไปทั้งหมดเป็น Married

### Original Dataset

Yes 398

No 213

### After preprocessing data

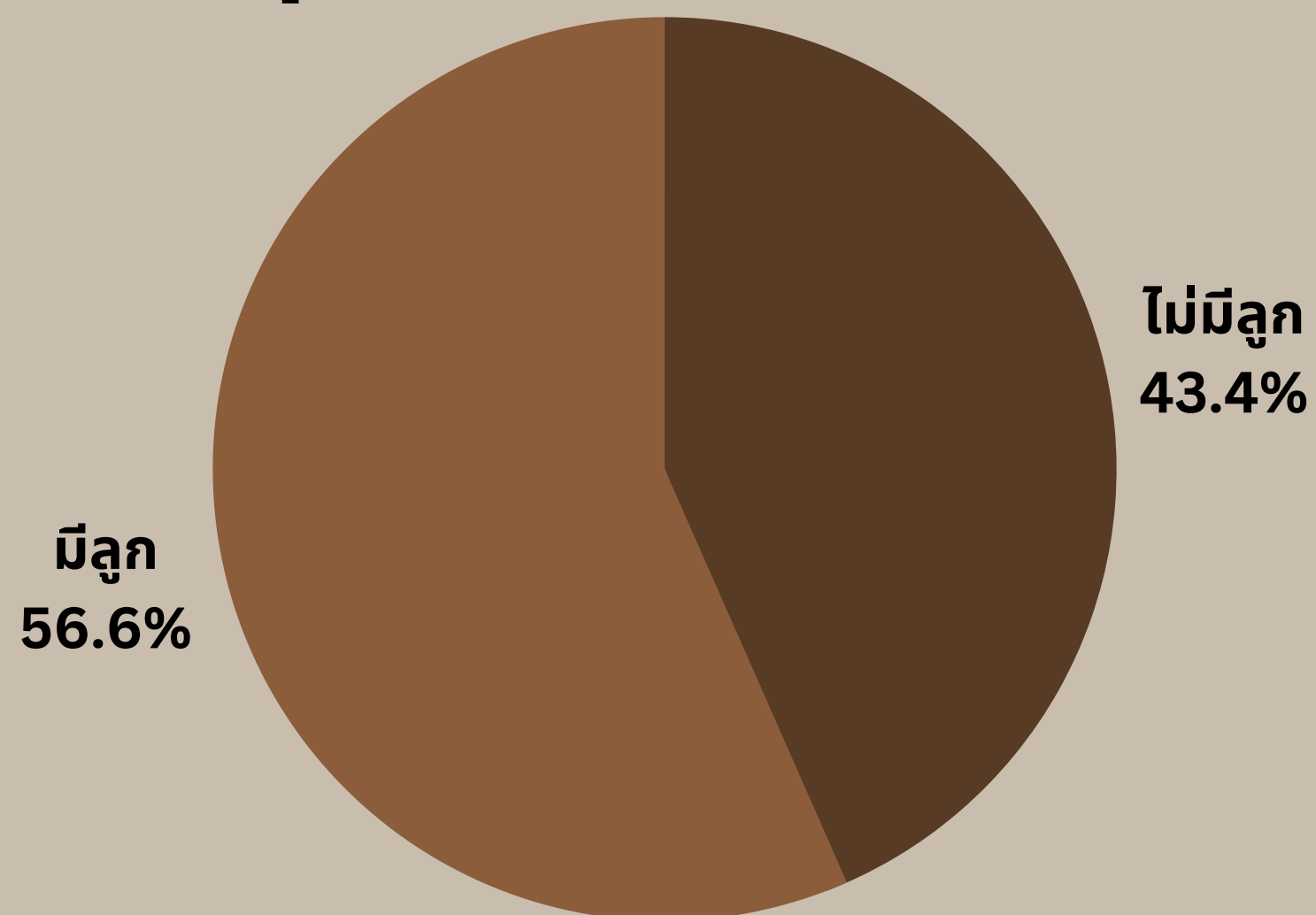
1 401

0 213

Missing value: 3

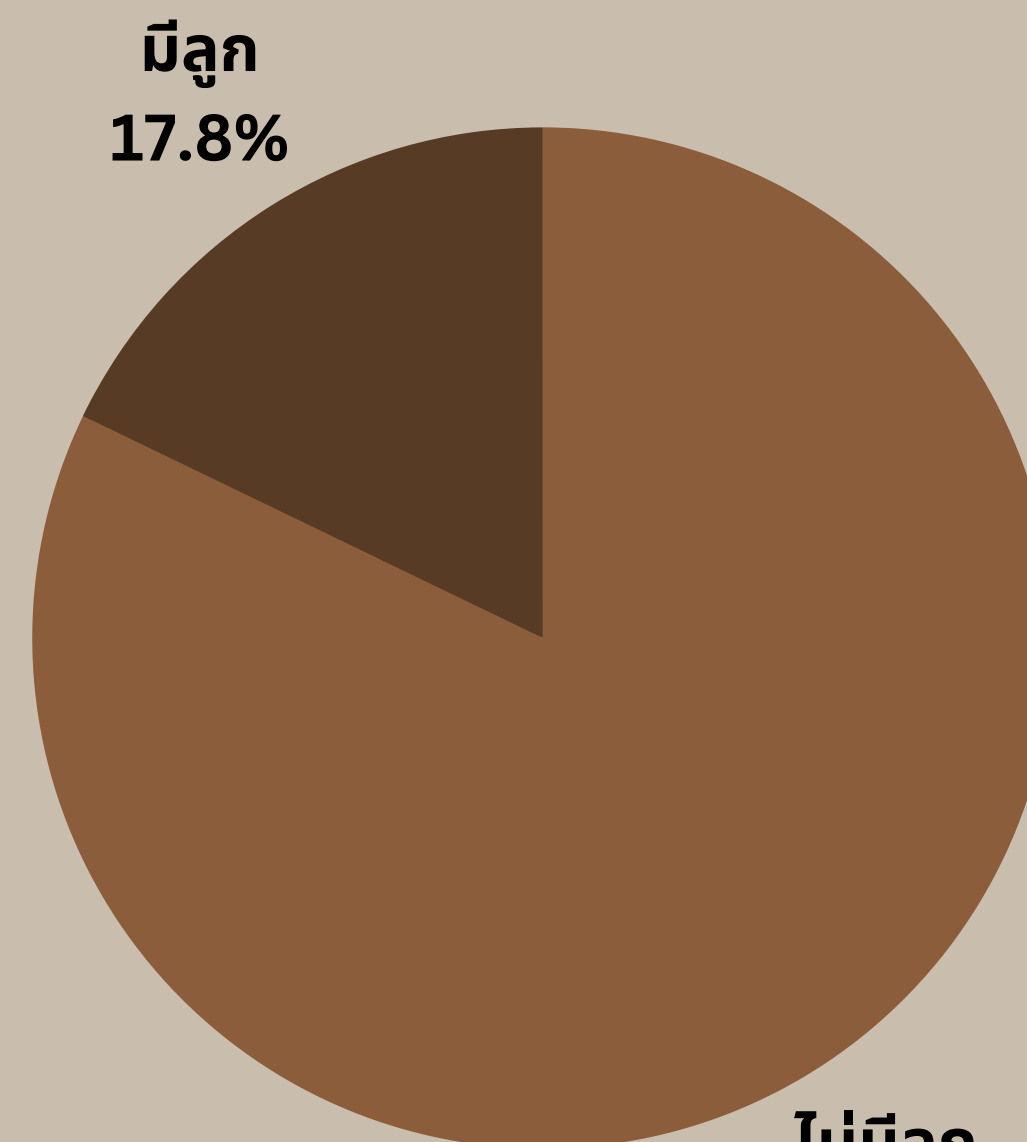
# Feature Engineering

## Dependents



แต่งงานแล้ว

จำนวนคนที่แต่งงานแล้วมีทั้งหมด 401 คน จำนวนคนที่มีลูกมี 227 คน คิดเป็น 56.6 %  
จำนวนคนที่ยังไม่ได้แต่งงานมีทั้งหมด 213 คน จำนวนคนที่ไม่ มีลูกเลย 175 คน คิดเป็น 82.15 %



ยังไม่ได้แต่งงาน

# Feature Engineering

## Dependents

ทางทีมจึงพิจารณาว่า Dependents คือ เด็กหรือลูกของผู้ขอสินเชื่อ โดย  
หากผู้ขอสินเชื่อไม่ได้แต่งงาน เราจะเติมค่าว่างนั้นด้วย 0  
หากผู้ขอสินเชื่อแต่งงานแล้ว เราจะเติมค่า Null ด้วย 1

### Original Dataset

0 345

1 102

2 101

3+ 51

### After preprocessing data

0 349

1 113

2 101

3+ 51

Missing value: 15

# Feature Engineering

## Self Employed

### Original Dataset

Yes 82

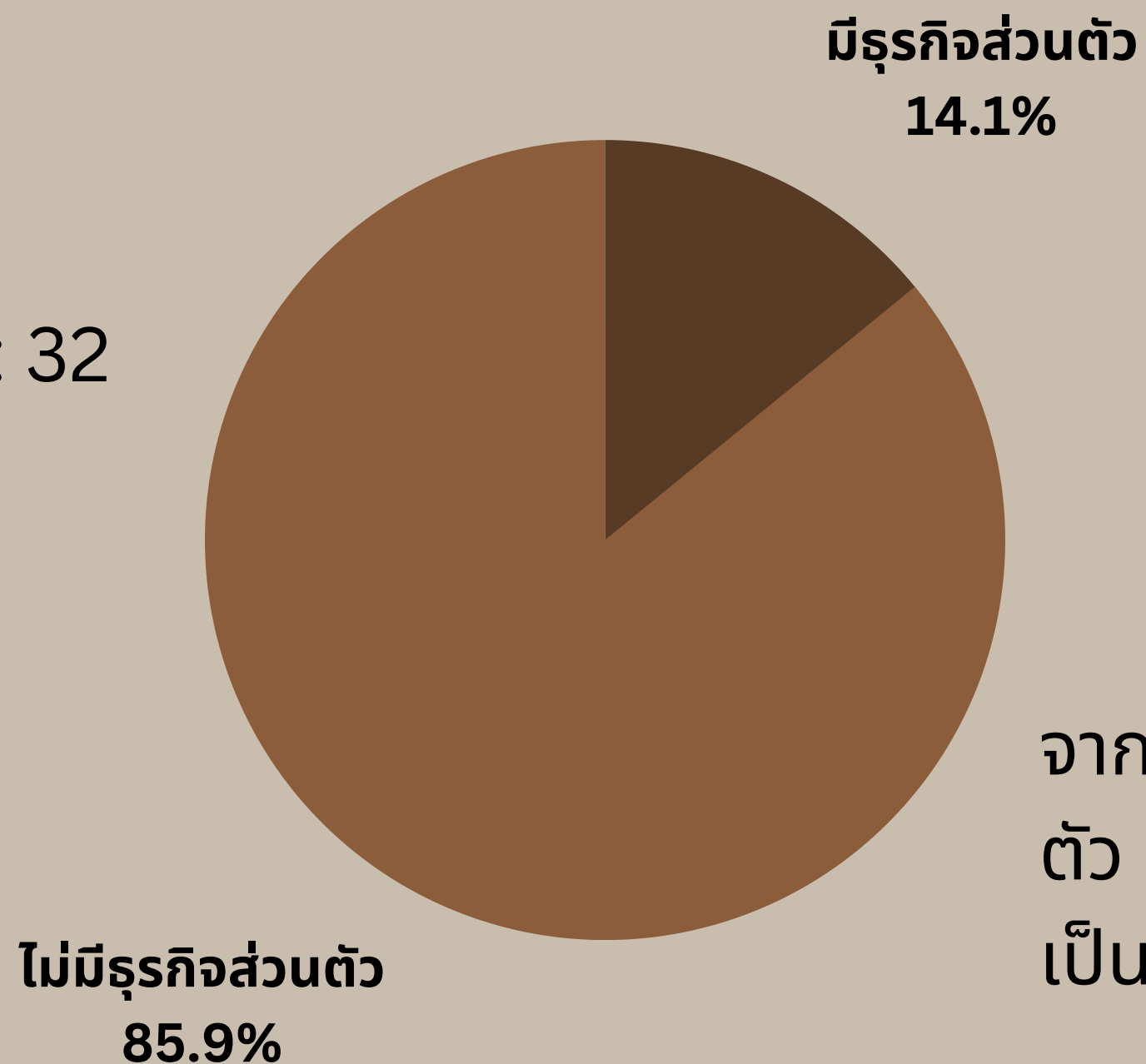
No 500

Missing value: 32

### After preprocessing data

Yes 82

No 532



จาก Dataset ข้อมูลลูกค้าส่วนใหญ่จะไม่มีธุรกิจส่วนตัว ทางทีมเลยพิจารณาว่า ข้อมูลส่วนที่หายไปจะให้ เป็น 'No' (ไม่มีธุรกิจส่วนตัว)

# Feature Engineering

## LoanAmount

Missing value: 22

Out[63]:

Loan_Status	N	Y
count	181.000000	411.000000
mean	151.220994	144.294404
std	85.862783	85.484607
min	9.000000	17.000000
25%	100.000000	100.000000
50%	129.000000	126.000000
75%	176.000000	161.000000
max	570.000000	700.000000

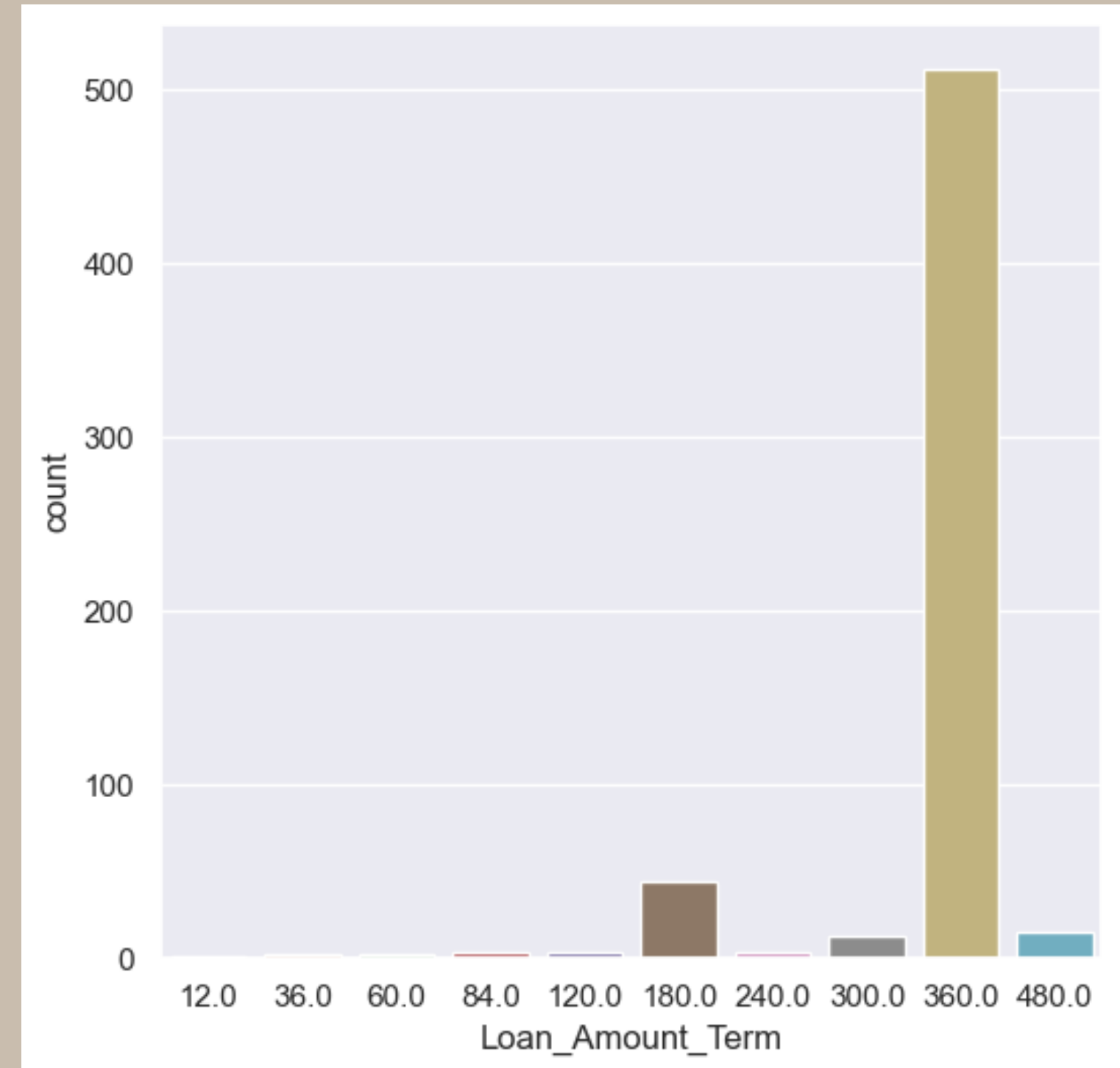
จาก Dataset ทางทีมได้พิจารณาจากผลของการอนุมัติสินเชื่อ (Loan\_Status) โดยให้ดูค่าสถิติต่าง ๆ ของ LoanAmount จะเห็นว่า ผลผ่านการอนุมัติจะมีค่าเฉลี่ยของ LoanAmount **151.22** ส่วนผลไม่ผ่านการอนุมัติจะมีค่าเฉลี่ย **144.29** เราจะค่าเฉลี่ยดังกล่าวมาเพิ่มในส่วนของ Missing value

# Feature Engineering

## Loan\_Amount\_Term

Missing value: 14

จาก Dataset ข้อมูลของ Loan\_Amount\_Term  
ส่วนใหญ่จะเป็น 360 ทางทีมจึงพิจารณาให้ค่าที่หายไป  
เป็น 360



# Feature Engineering

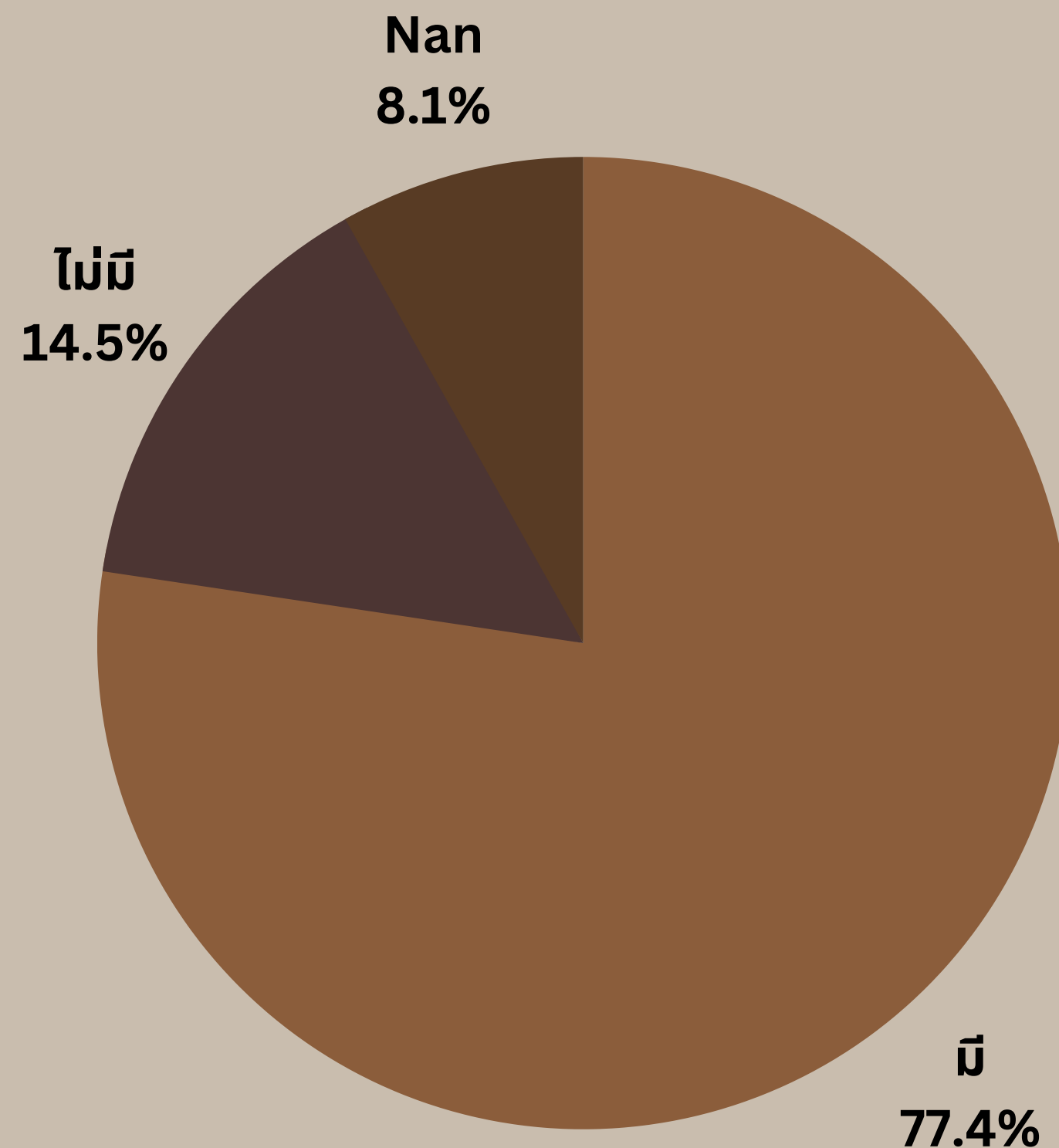
## Credit History

### Original Dataset

1 475 (มี)

0 89 (ไม่มี)

Missing value: 50



เนื่องจากผู้ขอสินเชื่อ 77.4 % มี Credit\_History ทางทีมจึงเติมค่า Null ด้วย 1.0

# Feature Engineering

## Property\_Area

Semiurban 233 (ชนบทกึ่งเมือง)

Urban 202 (เมือง)

Rural 179 (ชนบท)

เนื่องจากเป็นข้อมูลที่เป็น string ไม่สามารถนำมาเข้า Model ได้  
ทางทีมจึงทำ One-Hot Encoding หรือ Dummy Encoding  
เพื่อให้โมเดลสามารถจำแนกลักษณะของเมืองที่อยู่ได้

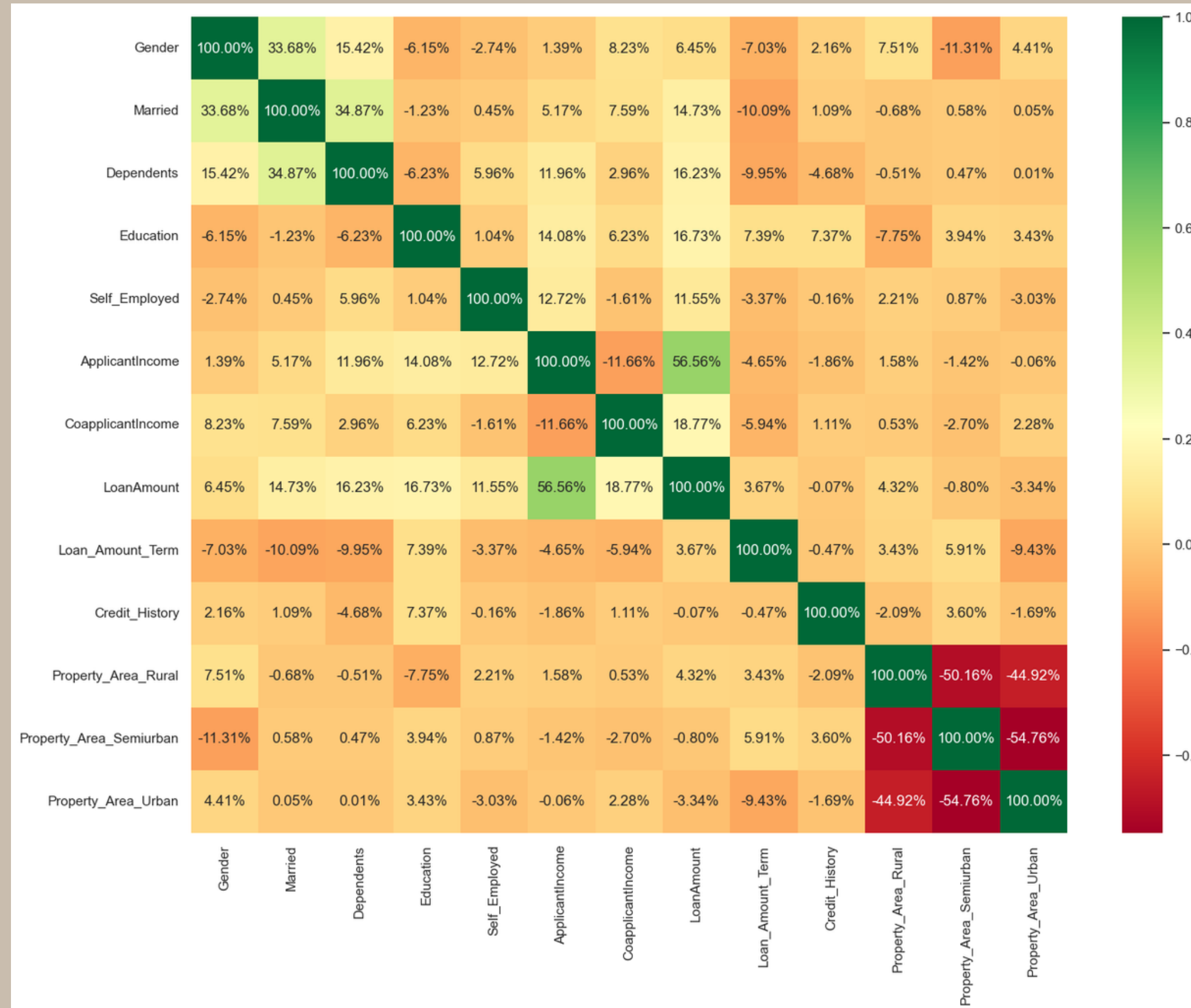
Out[82]:

	Rural	Semiurban	Urban
0	0	0	1
1	1	0	0
2	0	0	1
3	0	0	1
4	0	0	1
...	...	...	...
609	1	0	0
610	1	0	0
611	0	0	1
612	0	0	1
613	0	1	0



# Feature Selection

15



# Feature Selection

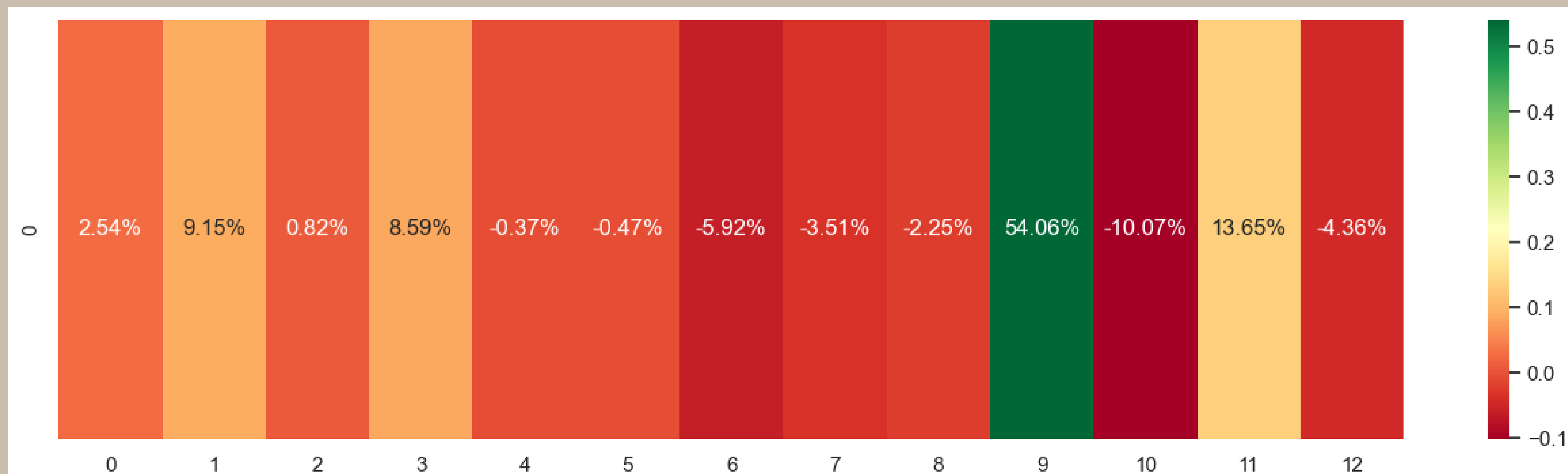
ความสัมพันธ์ระหว่าง Feature ทั้งหมดกับ ผลการอนุมัติสินเชื่อ

1. Gender 2. Married 3. Dependents 4. Education 5. Self\_Employed

6. ApplicantIncome 7. CoapplicantIncome 8. LoanAmount

9. Loan\_Amount\_Term **10. Credit\_History**

11. Property\_Area\_Rural 12. Property\_Area\_Semiurban 13. Property\_Area\_Urban



# Feature Selection

## Percentage Importance of each features with respect to Loan\_Status

Percentage Importance of each features with respect to Loan\_Status :

Gender	2.149608
Married	2.321261
Dependents	6.301363
Education	2.252524
Self_Employed	2.395319
ApplicantIncome	19.042403
CoapplicantIncome	11.468863
LoanAmount	18.220919
Loan_Amount_Term	4.905860
Credit_History	27.506254
Loan_Status	1.134685
Property_Area_Rural	1.425329
Property_Area_Semiurban	0.875611

### Feature ที่นำเข้า Model

- 1.credit\_History
- 2.LoanAmount
- 3.CoapplicantIncome
- 4.ApplicantIncome
- 5.Dependents

**ExtraTreesClassifier** คืออัลกอริทึมที่ใช้ในการสร้างและเรียนรู้ข้อมูลในรูปแบบของต้นไม้ตัดสินใจ (Decision Trees) หลาย ๆ ต้น และประเภทของเส้นทางตัดสินใจที่สุ่มขึ้นมาจากข้อมูล (Extremely Randomized Trees) ซึ่งเป็นวิธีการปรับปรุงที่คล้ายกับ Random Forest แต่ **ExtraTreesClassifier** สุ่มข้อมูลอย่างมากขึ้นในขั้นตอนของการแยกแยะข้อมูลที่แต่ละโหนด

# Model

## A Final overview of different kind of Model Based on their performance

ในส่วนของ Model ทางทีมได้ทำการทดสอบ Decision Tree โดยเพิ่มปัจจัย มา 2 อย่างคือ Standard Scaling และ PCA

Decision Tree Classifier Model  
without Standard Scaling and  
without PCA

Training Accuracy : 85.53 %  
Testing Accuracy : 85.36 %

Decision Tree Classifier Model  
with Standard Scaling and  
without PCA

Training Accuracy : 85.74 %  
Testing Accuracy : 85.36 %

Decision Tree Classifier Model  
with Standard Scaling and with  
PCA

Training Accuracy : 84.92 %  
Testing Accuracy : 83.73 %



# Model Comparision

1. Regression to  
predict null value

**Accuracy: 0.81**

2. Feature Engineering

**Accuracy: 0.85**



# Web application



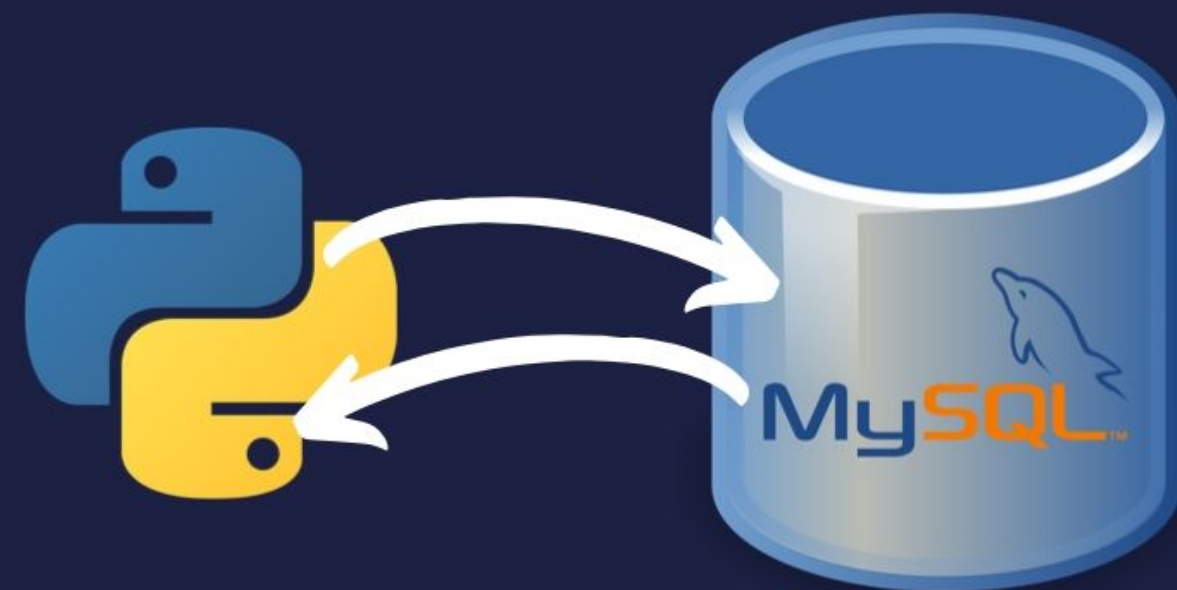
```
1 from flask import Flask, render_template, request, redirect, url_for, session, flash
2 from werkzeug.security import generate_password_hash, check_password_hash
3 import pymysql
```

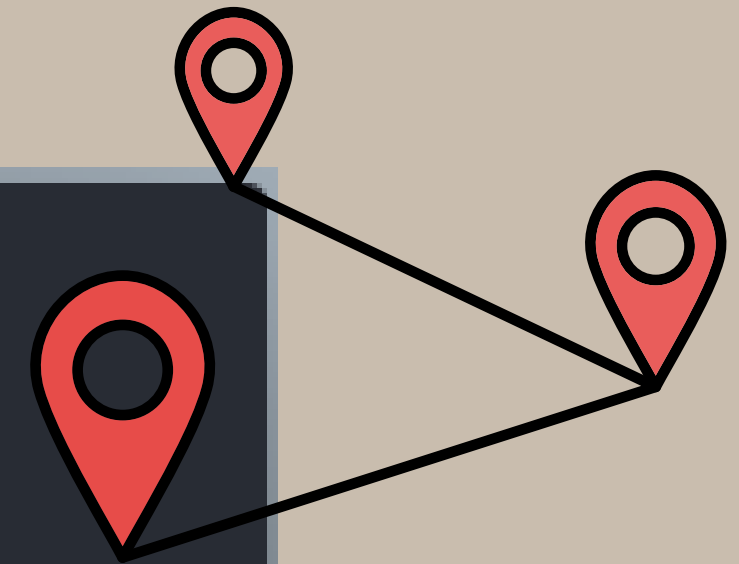


# Flask

web development,  
one drop at a time


PYTHON MYSQL DATABASE





```
1  @app.route("/")
2  def indexPage():
3      return render_template('NotLogin/index.html')
4
5  @app.route("/register")
6  def registerPage():
7      return render_template('NotLogin/register.html')
8
9  @app.route("/login")
10 def loginPage():
11     return render_template('NotLogin/login.html')
12
13 @app.route("/aboutUs")
14 def aboutUsPage():
15     return render_template('NotLogin/aboutUs.html')
```





```
1 @app.route("/register/actions", methods=['POST'])
2 def registerAction():
3     if request.method == "POST":
4         name = request.form['name']
5         email = request.form['email']
6         password = request.form['password']
7         confirmPassword = request.form['confirmPassword']
8         dateOfBirth = request.form['dateOfBirth']
9
10        if (password == confirmPassword):
11
12            # Hashed password
13            hashed_password = generate_password_hash(password, method='sha256')
14
15            conn = openConnection()
16            cur = conn.cursor()
17
18            sql = "INSERT INTO `employee`(`email`, `password`, `name`, `dateOfBirth`) VALUES (%s, %s, %s, %s)"
19            cur.execute(sql, (email, hashed_password, name, dateOfBirth))
20            conn.commit()
21            conn.close()
22
23            flash("Register Success", "success")
24            return redirect(url_for('loginPage'))
25        else:
26            flash("Password and Confirm Password Don't Match", "danger")
27            return redirect(url_for('registerPage'))
```





```

1 @app.route('/logout')
2 def logout():
3     session.pop('login', None)
4     return redirect(url_for('indexPage'))

```



LOGIN



 Username

 Password

LOGIN



```

1 @app.route("/login/actions", methods=['POST'])
2 def loginAction():
3     if request.method == "POST":
4         email = request.form['email']
5         password = request.form['password']
6         conn = openConnection()
7         cur = conn.cursor()
8         sql = "SELECT * FROM `employee` WHERE employee.email = %s"
9         cur.execute(sql, (email))
10        result = cur.fetchone()
11        conn.close()
12        if result == None :
13            flash("Don't have this User in database", "danger")
14            return redirect(url_for('loginPage'))
15        else :
16            if result and check_password_hash(result[2], password):
17                session['login'] = result[1]
18                return redirect(url_for('loginFormPage', id = result[0], name = result[3]))
19            else :
20                flash("Please check your password", "warning")
21                return redirect(url_for('loginPage'))

```

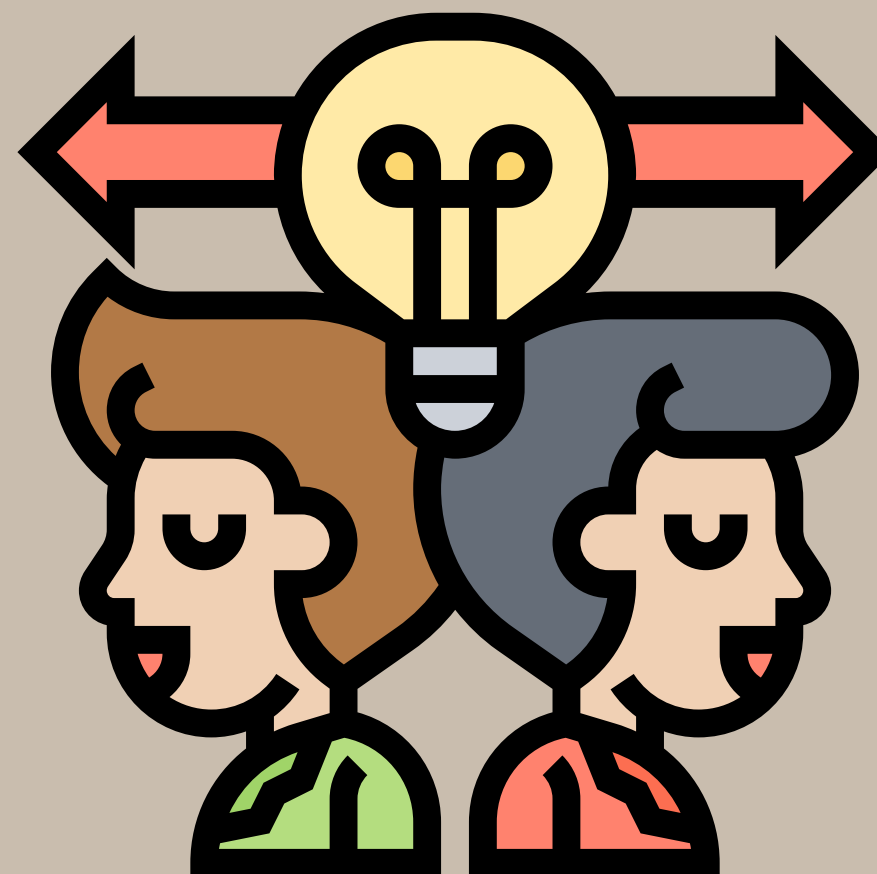
My Content

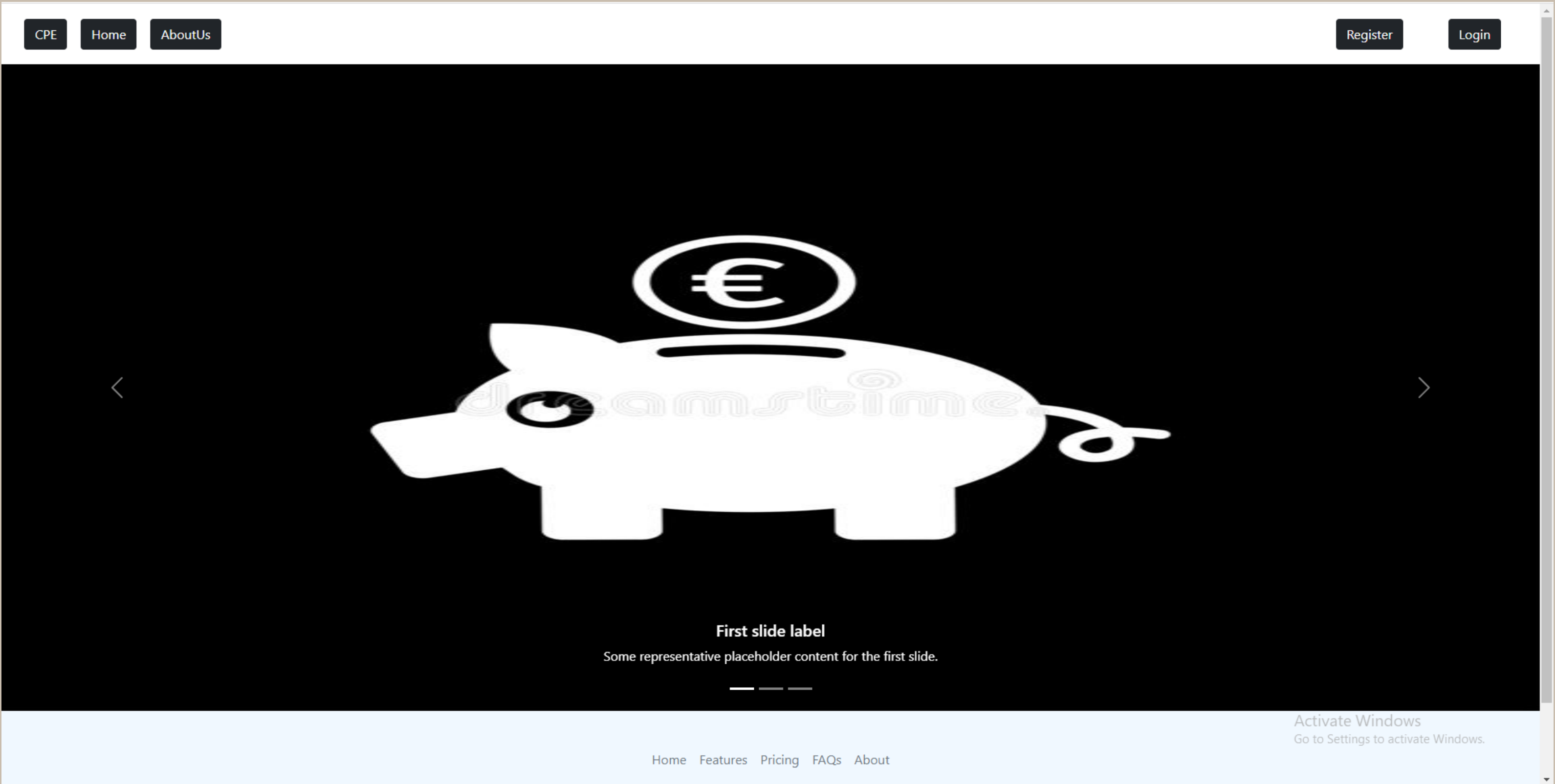
Profile

Settings


LOGOUT

```
1 @app.route("/<string:id>/name/<string:name>")
2 def loginFormPage(id, name):
3     if 'login' in session:
4         return render_template('Login/form.html', data_id = id, data_name = name)
5     else :
6         return redirect(url_for('indexPage'))
```





[CPE](#)[Home](#)[AboutUs](#)[Register](#)[Login](#)



## Register Page

Name

Email

Password


Confirm Password

Date of Birth

Activate Windows  
Go to Settings to activate Windows.

CPEHomeAboutUs

RegisterLogin



We are Computer Engineering

Please login to your account

Email address

Password

Log in

Don't have an account? Register

Data Warehouse and Data Mining

This project is part of the Data Warehouse and Data Mining course. Kasetsart University, Kamphaeng Saen Campus (KU.KPS), Faculty of EGINEERING at Kamphaeng Saen, Computer Engineering.

HomeFeaturesPricingFAQsAbout

Activate Windows  
Go to Settings to activate Windows.

Form

Name

Name

Surname

Surname

Age

Age

Gender

Select Gender

Phone number

Phone number

Married:

Yes

No

Dependents

Select Dependents

Education

Select Education

Self\_Employed:

Yes

No

Credit\_History:

Yes

No

ApplicantIncome

ApplicantIncome

CoapplicantIncome

CoapplicantIncome

Loan\_Amount

Loan amount in thousands \$

Loan\_Amount\_Term

Term of the loan in months

Property\_Area

Select Property\_Area

Submit

# Decide Page

Decide

4

Name : Test\_Name

Loan Amount : 1000.0

Loan Amount Term : 100000

Model\_Predict: Loan application denied.

• Customer does not meet credit history requirement.

• Customer does not meet income requirement.

• Customer does not meet coapplicant income requirement.

• Customer has too many dependents.

Denied

Approve

THANK YOU  
FOR YOUR  
ATTENTION

