

# Course 1 List of Modules and .csv's

th_income_tax.csv	
threshold min	rate

pct_dividends.csv	
	rate

th_employee_national_insurance.csv	
threshold min	rate

th_corporate_national_insurance.csv	
threshold min	rate

th_corporation_tax.csv	
threshold min	rate

allowance_change_income.csv	
	value

allowance_dividends_tax_free.csv	
	value

get_csv_for_dataframe	
tax_name (string)	
tax_year (string)	
CODE - 6 Lines	
OUT - DataFrame	

create_table	
threshold_table (DataFrame 2col)	
CODE - 4 Lines	
OUT - DataFrame - 3col	

marginal	
cash (integer)	
tax_table (DataFrame 3col)	
CODE - 12 Lines	
OUT - integer/float	

marginalise_table	
cash (integer)	
threshold_table (DataFrame 2col)	
create_table	
marginal	
OUT - DataFrame - 3col	

interpret_single_df_value	
df (DataFrame 1col,1row)	
OUT - integer	

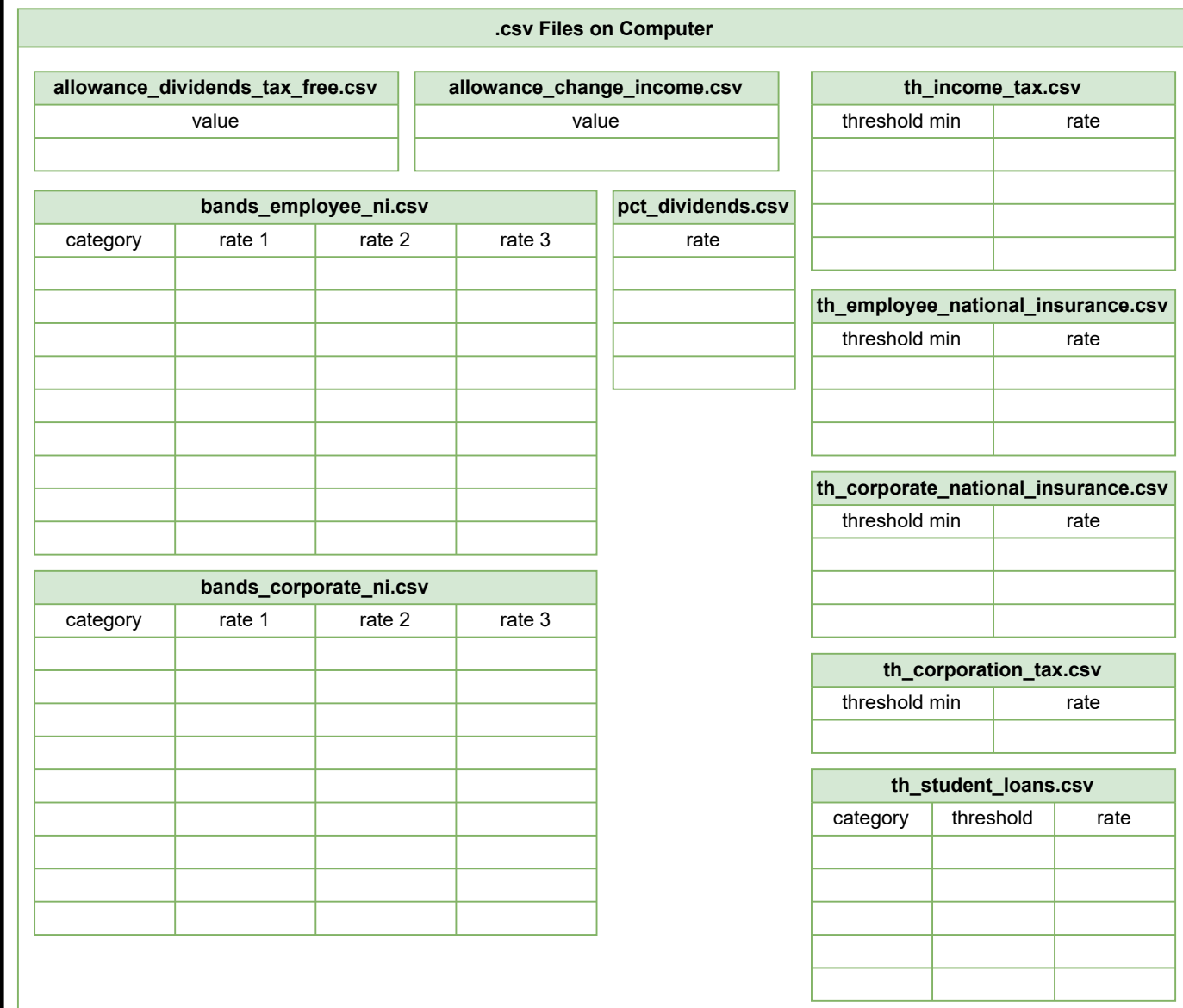
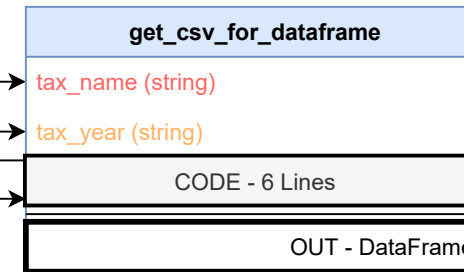
replace_threshold_rates	
threshold_table (DataFrame 2col)	
rates (DataFrame 1col)	
CODE - 3 Lines	
OUT - DataFrame - 2col	

# find\_csv\_for\_dataframe

```
def get_csv_for_dataframe(tax_name, tax_year):
```

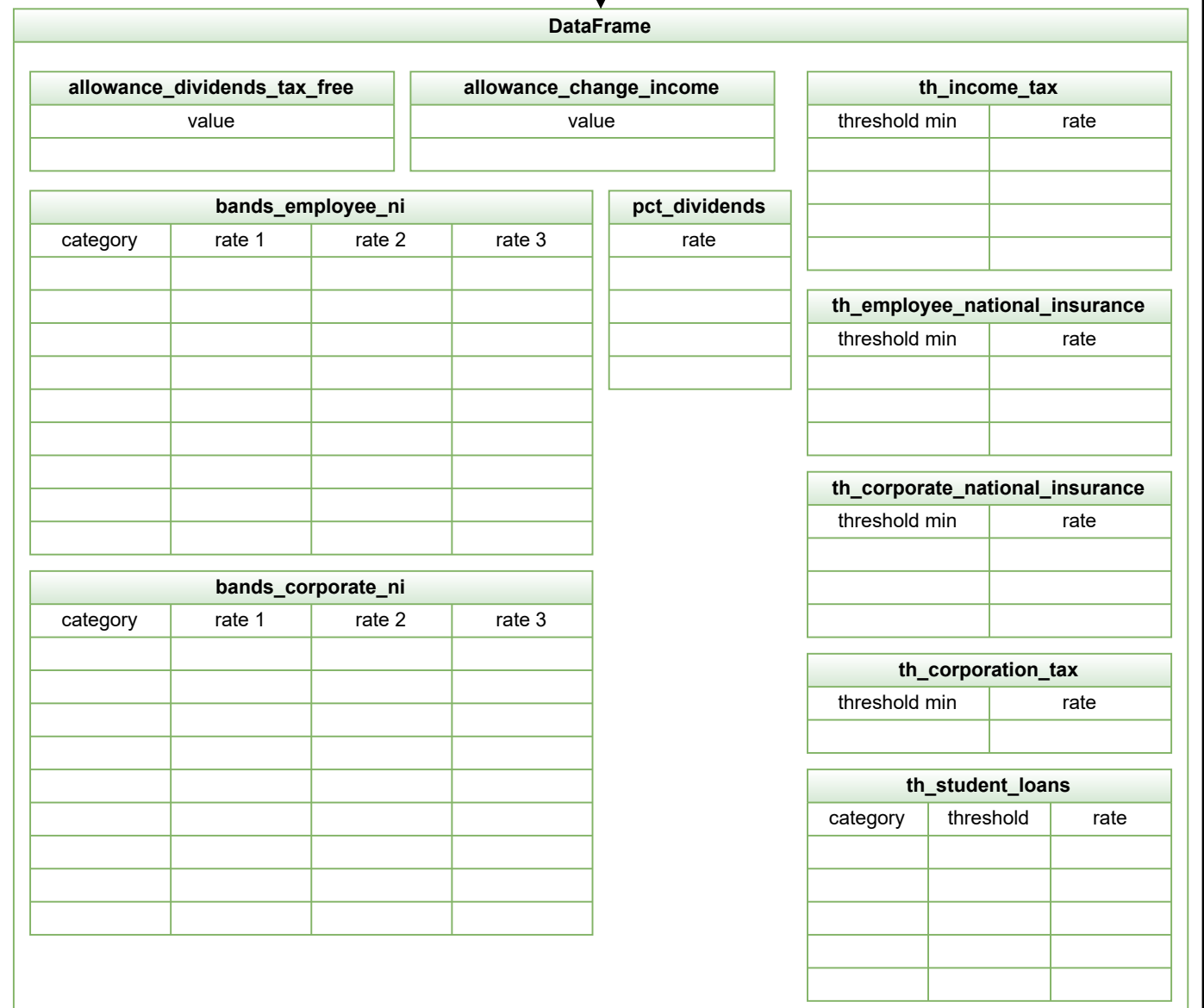
string
'2019/2020'
'2020/2021'
'2021/2022'
'2022/2023'

string
'income tax'
'employee ni'
'corporate ni'
'dividend rates'
'student loans'
'corporation tax'
'employee ni bands'
'corporate ni bands'
'high income threshold'
'dividend tax free allowance'



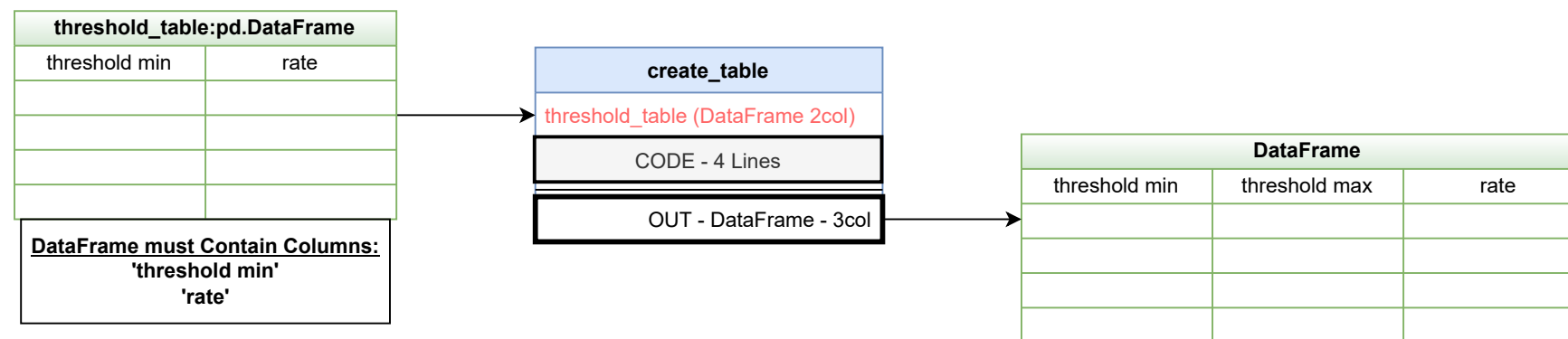
Find File

Import Data from File



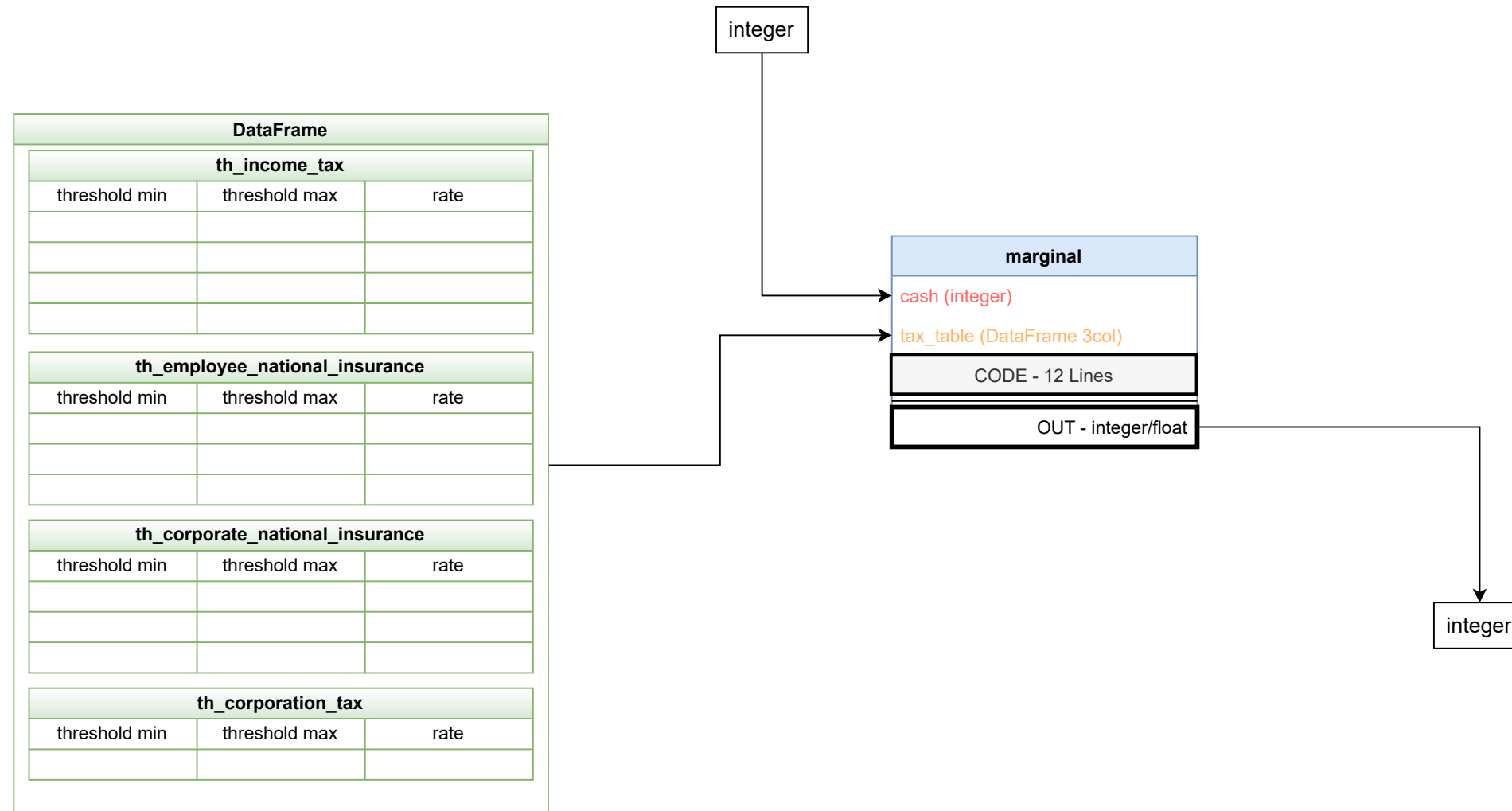
# create\_table

```
def create_table(threshold_table):
```



# marginal

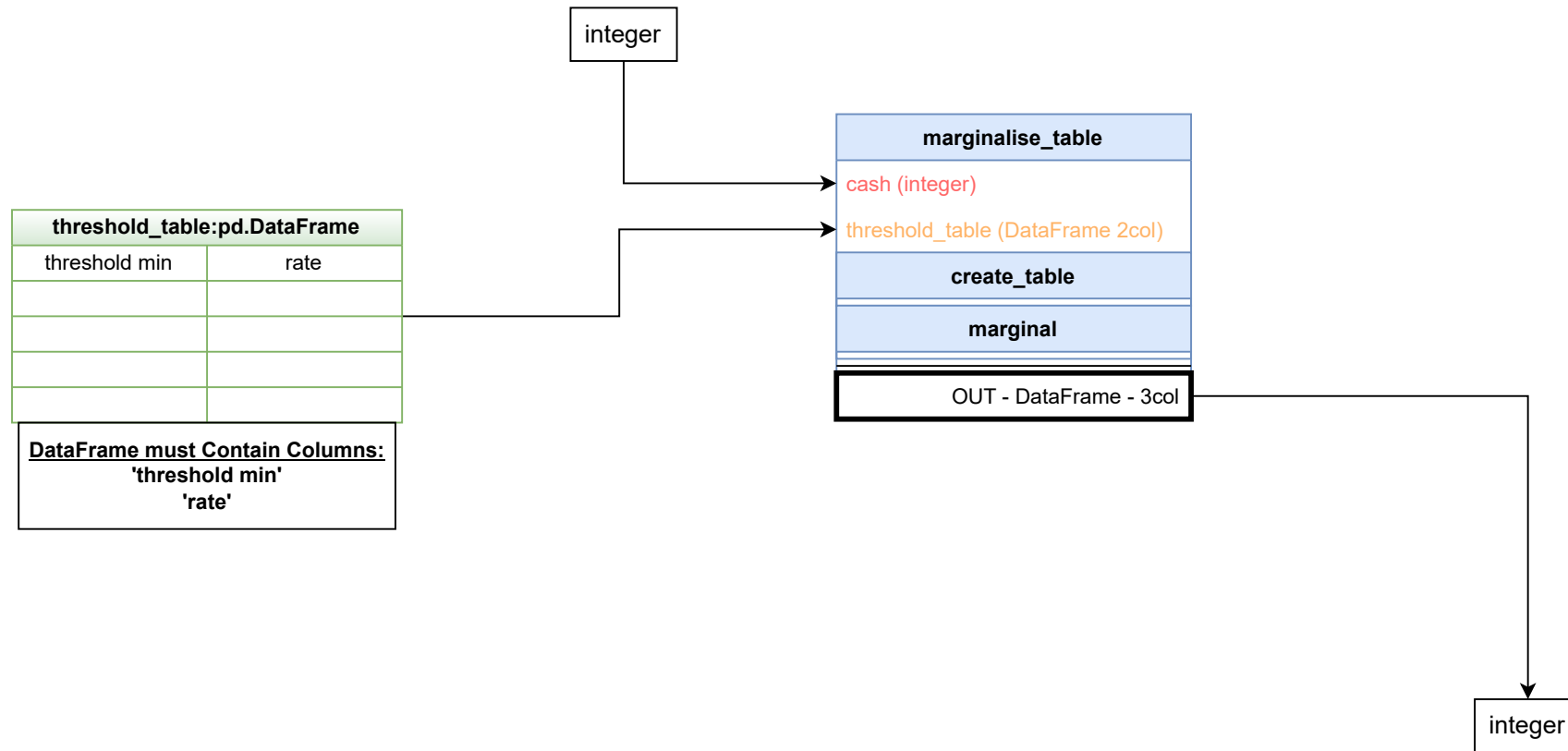
```
def marginal(cash,tax_table):
```



# marginalise

```
def marginalise(cash, threshold_table):
```

get_csv_for_dataframe
tax_name (string)
tax_year (string)
CODE - 6 Lines
OUT - DataFrame



# Full List of Modules and .csv's

th_income_tax.csv	
threshold min	rate

pct_dividends.csv
rate

th_employee_national_insurance.csv	
threshold min	rate

th_corporate_national_insurance.csv	
threshold min	rate

th_corporation_tax.csv	
threshold min	rate

th_student_loans.csv		
category	threshold min	rate

bands_employee_ni.csv			
category	rate 1	rate 2	rate 3

bands_corporate_ni.csv			
category	rate 1	rate 2	rate 3

income_tax
salary (integer)
tax_code (string)
tax_year (string)
<b>get_csv_for_dataframe</b>
<b>interpret_tax_code_allowance</b>
<b>adjust_allowance</b>
<b>marginalise</b>
OUT - Number

corporate_tax
gross_profit (integer)
tax_year (string)
<b>get_csv_for_dataframe</b>
<b>marginalise</b>
OUT - Number

allowance_change_income.csv
value

allowance_dividends_tax_free.csv
value

get_csv_for_dataframe
tax_name (string)
tax_year (string)
CODE - 6 Lines
OUT - DataFrame

create_table
threshold_table (DataFrame 2col)
CODE - 4 Lines
OUT - DataFrame - 3col

marginal
cash (integer)
tax_table (DataFrame 3col)
CODE - 12 Lines
OUT - integer/float

marginalise_table
cash (integer)
threshold_table (DataFrame 2col)
<b>create_table</b>
<b>marginal</b>
OUT - DataFrame - 3col

corporate_ni
salary (integer)
tax_year (string)
tax_code (string)
<b>get_csv_for_dataframe</b>
<b>interpret_tax_code_ni_band</b>
<b>find_csv_for_dataframe</b>
<b>marginalise</b>
<b>adjust_ni_band</b>
<b>marginalise</b>
OUT - Number

dividend_tax
salary (integer)
dividend (string)
tax_code (string)
tax_year (string)
<b>get_csv_for_dataframe</b>
<b>interpret_tax_code_allowance</b>
<b>adjust_allowance</b>
<b>find_csv_for_dataframe</b>
<b>replace_threshold_rates</b>
<b>find_csv_for_dataframe</b>
<b>interpret_single_df_value</b>
<b>create_dividend_table</b>
CODE - 1 Line
<b>marginalise</b>
OUT - Number

student_loans
plan (string)
cash (integer)
tax_year (string)
<b>get_csv_for_dataframe</b>
<b>marginalise</b>
OUT - Number

interpret_single_df_value
df (DataFrame 1col,1row)
OUT - integer

replace_threshold_rates
threshold_table (DataFrame 2col)
rates (DataFrame 1col)
CODE - 3 Lines
OUT - DataFrame - 2col

salary_taxes
salary (integer)
tax_code (string)
tax_year (string)
student_loan_plan (string)
student_loan_second_plan (string)
<b>employee_ni</b>
<b>income_tax</b>
<b>corporate_ni</b>
<b>student_loans</b>
CODE - 4 Lines
OUT - DataFrame

adjust_allowance
salary (integer)
tax_allowance (integer)
tax_table (DataFrame - 2col)
tax_year (string)
CODE - 4 Lines
<b>get_csv_for_dataframe</b>
<b>interpret_single_df_value</b>
CODE - 10 Lines
OUT - DataFrame - 2col

interpret_tax_code_allowance
tax_code (string)
CODE - 3 Lines
OUT - integer

interpret_tax_code_ni_band
tax_code (string)
CODE - 1 Line
OUT - string

ltd_owner_full_take
turnover (integer)
expenses (integer)
salary (integer)
tax_code (string)
tax_year (string)
student_loan_plan (string)
student_loan_second_plan (string)
<b>salary_taxes</b>
CODE - 2 Lines
<b>corporate_tax</b>
CODE - 2 Lines
<b>dividend_tax</b>
CODE - 1 Lines
<b>student_loans</b>
<b>student_loans</b>
CODE - 7 Lines
OUT - DataFrame - 1 Row

adjust_ni_band
tax_table (DataFrame - 2col)
ni_band_table (DataFrame - 4col)
ni_band (string)
CODE - 9 Lines
OUT - DataFrame - 2col

create_dividend_table
salary (integer)
tax_free_amount (integer)
tax_table (DataFrame - 3col)
CODE - 9 Lines
OUT - DataFrame - 2 Column

create_student_loan_table
plan (string)
plan_table (DataFrame - 3col)
CODE - 3 Lines
OUT - DataFrame - 2col

iterate_salaries_ltd_take
turnover (integer)
expenses (integer)
min_salary (integer)
max_salary (integer)
tax_code (string)
tax_year (string)
student_loan_plan (string)
student_loan_second_plan (string)
iteration_step (string)
CODE - 2 Lines
<b>ltd_owner_full_take</b>
CODE - 2 Lines
OUT - DataFrame - Many Rows

optimal_take
options (DataFrame)
CODE - 2 Lines
OUT - DataFrame - 1row

iterate_ltd_owner_full_take
turnover (integer)
expenses (integer)
tax_code (string)
tax_year (string)
student_loan_plan (string)
student_loan_second_plan (string)
CODE - 1 Line
<b>iterate_salaries_ltd_take</b>
OUT - DataFrame - Many Rows

optimise_ltd_owner_full_take
turnover (integer)
expenses (integer)
tax_code (string)
tax_year (string)
student_loan_plan (string)
student_loan_second_plan (string)
<b>iterate_salaries_ltd_take</b>
<b>optimal_take</b>
OUT - DataFrame - 1 Row

optimise_lite_ltd_owner_full_take
turnover (integer)
expenses (integer)
tax_code (string)
tax_year (string)
student_loan_plan (string)
student_loan_second_plan (string)
iteration_step (string)
CODE - 1 Lines
<b>iterate_salaries_ltd_take</b>
<b>ltd_owner_full_take</b>
CODE - 10 Lines
<b>iterate_salaries_ltd_take</b>
<b>optimal_take</b>
OUT - DataFrame - 1 Row

# UK Tax for Limited Company Business Owners

