

def main(self):

- Calls **def select_docs(self):**
 - Select the docx (Exam Bill Demo.docx).
- Calls **def self.select_sample_excel(self):**
 - Select the sample excel (sample.xlsx).
- Calls **def generate_excel_from_docs(self):**
 - Calls **self.extract_data_from_docx()**
 - Calls **def self.extract_department_line()**
 - Get *self.dept*
 - Calls **self.dept_translate_to_bengali(self.dept.lower())**
 - Translate *dept* (Mother Department)
 - Calls **def self.extract_year_and_term()**
 - Get *self.year*
 - Calls **self.convert_year_term_suffixes_to_bengali(self.year)**
 - Translate *year*
 - Get *self.term*
 - Calls **self.convert_year_term_suffixes_to_bengali(self.term)**
 - Translate *term*
 - Get *self.AD*
 - Calls **self.convert_year_term_suffixes_to_bengali(self.term)**
 - Translate *AD*
 - Write the global information into sample.xlsx
 - Generate all_tables.xlsx based on sample.xlsx (It includes all the tables from Exam Bill Demo.docx file).
 - Calls **def process_first_excel(self):**
 - Create excel file for every teacher.
 - Extract *name, designation, department*(teachersdepartment)
 - Calls **def print_matching_value_for_file(self, new_file, name, designation, department)**
 - Get some information
 - Calls **self.translate_to_bengali(name)**
 - Check if **def self.should_skip_translation(part)**
 - Return true if part(text) is special, else return false
 - Translate the name
 - Calls **self.translate_to_bengali(designation)**
 - Check if **def self.should_skip_translation(part)**
 - Return true if part(text) is special, else return false
 - Translate the designation
 - Calls **self.dept_translate_to_bengali(department.lower())**
 - Translate dept (department)
 - Get the bills for the corresponding teacher from the all_tables.xlsx.
 - Determine which bills is needed to write in which cell.
 - Calls **def english_to_bengali_number_in_words(self, english_number):**
 - Translate *amount*
 - Calls **self.clear_labels()**