Extracting Total Experience from Job Descriptions

📌 **Objective**

You are provided with an Excel sheet containing **200 Job Descriptions (JDs)**. Your task is to:

1. **Research and choose 5–6 models or techniques** to extract the **total experience required for that job** from each JD.
2. Write a code which implements all model in single code to give comparison of the output of each model extract total experience.
3. **Compare the results from each model** and output them in an Excel sheet.
4. **Analyze which model performs the best** based on consistency, logic, or accuracy (as per your understanding).
5. **Submit your recommendation** of the best model with justification.

✅ **Deliverables**

Your submission must include:

1. 📁 **Code (well-commented)** – which does the following:
   - Reads the input Excel file with 200 JDs
   - Applies each selected model/technique to extract the "total experience required"
   - Outputs the results from all models into an Excel file (each row should show results from all models)
   - Optionally includes performance comparison logic if ground truth is available
2. 📊 **Output Excel Sheet** – containing:
   - Original JD
   - Extracted experience values from each model (in separate columns)
3. 🏆 **Best Model Recommendation** – A short write-up (Markdown or PDF) answering:
   - Which model gave the best results?
   - Why do you think it performed best?
   - What were the challenges you faced in comparison?

🔍 Guidelines for Research & Model Selection

You are free to use any combination of:

- **Traditional NLP libraries** (e.g., spaCy, NLTK)
- **Transformer models** (e.g., BERT, DistilBERT, RoBERTa)
- **Question-answering models** (e.g., deepset/bert-base-cased-squad2)
- **Few-shot or zero-shot models** (e.g., TARS, GPT-like models)
- **Custom fine-tuned models** (optional, if time permits)

PS: Do not use Regex