

BaseSecuenciales.py

Modelos de base de datos para las rúbricas de ejercicios secuenciales

```
1 """
2 Modelos de base de datos para las rúbricas de ejercicios secuenciales.
3 Utiliza SQLAlchemy para la conexión con PostgreSQL.
4 """
5
6 from sqlalchemy import create_engine, Column, Integer, String, Text, ForeignKey, Float
7 from sqlalchemy.ext.declarative import declarative_base
8 from sqlalchemy.orm import relationship, sessionmaker
9
10 Base = declarative_base()
11
12
13 class Lenguaje(Base):
14     """Tabla de lenguajes de programación"""
15     __tablename__ = 'lenguaje'
16
17     id = Column(Integer, primary_key=True, autoincrement=True)
18     nombre = Column(String(50), nullable=False, unique=True) # Python, Java, JavaScript, etc.
19
20     # Relación con ejercicios
21     ejercicios = relationship("Ejercicio", back_populates="lenguaje")
22
23
24 class EscalaCalificacion(Base):
25     """Tabla para la escala de calificación general (Excelente, Bueno, Regular, etc.)"""
26     __tablename__ = 'escala_calificacion'
27
28     id = Column(Integer, primary_key=True, autoincrement=True)
29     nombre = Column(String(50), nullable=False, unique=True) # Excelente, Bueno, Regular, etc.
30     puntos = Column(Integer, nullable=False) # 4, 3, 2, 1, 0
31     descripcion = Column(String(200)) # Cumple todos los criterios correctamente, etc.
32
33
34 class Ejercicio(Base):
35     """Tabla principal de ejercicios"""
36     __tablename__ = 'ejercicio'
37
38     id = Column(Integer, primary_key=True, autoincrement=True)
39     numero = Column(Integer, nullable=False, unique=True) # 1, 2, 3... 10
40     titulo = Column(String(100), nullable=False) # Hola Mundo, Saludo Personalizado, etc.
41     objetivo = Column(Text, nullable=False) # Descripción del objetivo del ejercicio
42     puntaje_maximo = Column(Integer, nullable=False) # 12, 16, 24, etc.
43     lenguaje_id = Column(Integer, ForeignKey('lenguaje.id'), nullable=True) # FK al lenguaje de programación
44
45     # Relación con lenguaje
46     lenguaje = relationship("Lenguaje", back_populates="ejercicios")
47
48     # Relación con criterios
49     criterios = relationship("Criterio", back_populates="ejercicio", cascade="all, delete-orphan")
50
51
52 class Criterio(Base):
53     """Tabla de criterios de evaluación por ejercicio"""
54     __tablename__ = 'criterio'
55
56     id = Column(Integer, primary_key=True, autoincrement=True)
57     ejercicio_id = Column(Integer, ForeignKey('ejercicio.id'), nullable=False)
58     nombre = Column(String(100), nullable=False) # Uso correcto de print(), Mensaje correcto, etc.
59     orden = Column(Integer, nullable=False) # Orden del criterio dentro del ejercicio
60
61     # Relación con ejercicio
62     ejercicio = relationship("Ejercicio", back_populates="criterios")
63
64     # Relación con niveles de desempeño
65     niveles = relationship("NivelDesempeno", back_populates="criterio", cascade="all, delete-orphan")
66
67
68 class NivelDesempeno(Base):
69     """Tabla de niveles de desempeño por criterio (Excelente, Bueno, Regular, Insuficiente)"""
70     __tablename__ = 'nivel_desempeno'
71
72     id = Column(Integer, primary_key=True, autoincrement=True)
73     criterio_id = Column(Integer, ForeignKey('criterio.id'), nullable=False)
74     escala_id = Column(Integer, ForeignKey('escala_calificacion.id'), nullable=False)
75     descripcion = Column(Text, nullable=False) # Descripción específica del nivel
76
77     # Relaciones
78     criterio = relationship("Criterio", back_populates="niveles")
79     escala = relationship("EscalaCalificacion")
80
81
82 class BuenaPractica(Base):
83     """Tabla de buenas prácticas de código"""
84     __tablename__ = 'buena_practica'
85
```

```

86     id = Column(Integer, primary_key=True, autoincrement=True)
87     descripcion = Column(Text, nullable=False)
88
89
90 class ErrorComun(Base):
91     """Tabla de errores comunes a penalizar"""
92     __tablename__ = 'error_comun'
93
94     id = Column(Integer, primary_key=True, autoincrement=True)
95     descripcion = Column(Text, nullable=False)
96
97
98 # Configuración de conexión a PostgreSQL
99 DATABASE_URL = "postgresql://usuario:contraseña@localhost:5432/rubricas_secuenciales"
100
101
102 def get_engine(database_url=DATABASE_URL):
103     """Crea y retorna el engine de SQLAlchemy"""
104     return create_engine(database_url)
105
106
107 def create_tables(engine):
108     """Crea todas las tablas en la base de datos"""
109     Base.metadata.create_all(engine)
110
111
112 def get_session(engine):
113     """Crea y retorna una sesión de base de datos"""
114     Session = sessionmaker(bind=engine)
115     return Session()
116
117
118 def insertar_datos_iniciales(session):
119     """Inserta los datos iniciales de la rúbrica"""
120
121     # Insertar lenguajes de programación
122     lenguajes = [
123         Lenguaje(nombre="Python"),
124         Lenguaje(nombre="Java"),
125         Lenguaje(nombre="JavaScript"),
126         Lenguaje(nombre="C"),
127         Lenguaje(nombre="C++"),
128         Lenguaje(nombre="C#"),
129         Lenguaje(nombre="PHP"),
130         Lenguaje(nombre="Ruby"),
131         Lenguaje(nombre="Go"),
132         Lenguaje(nombre="Rust"),
133     ]
134     session.add_all(lenguajes)
135     session.commit()
136
137     # Diccionario de lenguajes para referencia
138     lenguaje_dict = {l.nombre: l for l in session.query(Lenguaje).all()}
139
140     # Insertar escala de calificación
141     escalas = [
142         EscalaCalificacion(nombre="Excelente", puntos=4, descripcion="Cumple todos los criterios correctamente"),
143         EscalaCalificacion(nombre="Bueno", puntos=3, descripcion="Cumple la mayoría de los criterios con errores menores"),
144         EscalaCalificacion(nombre="Regular", puntos=2, descripcion="Cumple parcialmente, presenta errores significativos"),
145         EscalaCalificacion(nombre="Insuficiente", puntos=1, descripcion="No cumple los criterios mínimos"),
146         EscalaCalificacion(nombre="No presentado", puntos=0, descripcion="No entregó o no funciona"),
147     ]
148     session.add_all(escalas)
149     session.commit()
150
151     # Diccionario de escalas para referencia
152     escala_dict = {e.nombre: e for e in session.query(EscalaCalificacion).all()}
153
154     # Datos de ejercicios con sus criterios
155     ejercicios_data = [
156         {
157             "numero": 1,
158             "titulo": "Hola Mundo",
159             "objetivo": "Imprimir 'Hola Mundo!' por pantalla",
160             "puntaje_maximo": 12,
161             "criterios": [
162                 {
163                     "nombre": "Uso correcto de print()",
164                     "niveles": {
165                         "Excelente": "Usa print() con sintaxis perfecta",
166                         "Bueno": "Usa print() con error menor de sintaxis",
167                         "Regular": "Intenta usar print() pero con errores",
168                         "Insuficiente": "No usa print()"
169                     }
170                 },
171                 {
172                     "nombre": "Mensaje correcto",
173                     "niveles": {
174                         "Excelente": "Imprime exactamente 'Hola Mundo!'",
175                         "Bueno": "Imprime mensaje similar con variación menor",
176                         "Regular": "Imprime mensaje diferente",
177                         "Insuficiente": "No imprime nada"
178                     }
179                 }
180             ]
181         }
182     ]

```

```

179         },
180         {
181             "nombre": "Ejecución",
182             "niveles": {
183                 "Excelente": "Programa ejecuta sin errores",
184                 "Bueno": "Ejecuta con advertencias",
185                 "Regular": "Ejecuta con errores menores",
186                 "Insuficiente": "No ejecuta"
187             }
188         }
189     ],
190 },
191 {
192     "numero": 2,
193     "titulo": "Saludo Personalizado",
194     "objetivo": "Pedir nombre al usuario e imprimir saludo personalizado",
195     "puntaje_maximo": 16,
196     "criterios": [
197         {
198             "nombre": "Uso de input()",
199             "niveles": {
200                 "Excelente": "Usa input() correctamente con mensaje claro",
201                 "Bueno": "Usa input() sin mensaje descriptivo",
202                 "Regular": "Usa input() con errores",
203                 "Insuficiente": "No usa input()"
204             }
205         },
206         {
207             "nombre": "Almacenamiento en variable",
208             "niveles": {
209                 "Excelente": "Variable con nombre descriptivo",
210                 "Bueno": "Variable con nombre poco claro",
211                 "Regular": "Variable mal declarada",
212                 "Insuficiente": "No usa variable"
213             }
214         },
215         {
216             "nombre": "Uso de f-string",
217             "niveles": {
218                 "Excelente": "Usa f-string correctamente f'Hola {nombre}!!",
219                 "Bueno": "Usa concatenación con +",
220                 "Regular": "Imprime sin incorporar el nombre",
221                 "Insuficiente": "No imprime saludo"
222             }
223         },
224         {
225             "nombre": "Formato de salida",
226             "niveles": {
227                 "Excelente": "Salida exacta al formato pedido",
228                 "Bueno": "Salida con variación menor",
229                 "Regular": "Salida incompleta",
230                 "Insuficiente": "Sin salida"
231             }
232         }
233     ]
234 },
235 {
236     "numero": 3,
237     "titulo": "Datos Personales",
238     "objetivo": "Solicitar nombre, apellido, edad y lugar de residencia, e imprimir oración completa",
239     "puntaje_maximo": 16,
240     "criterios": [
241         {
242             "nombre": "Solicitud de 4 datos",
243             "niveles": {
244                 "Excelente": "Solicita los 4 datos con mensajes claros",
245                 "Bueno": "Solicita los 4 datos sin mensajes claros",
246                 "Regular": "Solicita menos de 4 datos",
247                 "Insuficiente": "No solicita datos"
248             }
249         },
250         {
251             "nombre": "Nombres de variables",
252             "niveles": {
253                 "Excelente": "Variables descriptivas (nombre, apellido, edad, lugar_de_residencia)",
254                 "Bueno": "Variables entendibles pero no óptimas",
255                 "Regular": "Variables confusas",
256                 "Insuficiente": "Sin variables o mal usadas"
257             }
258         },
259         {
260             "nombre": "Construcción de oración",
261             "niveles": {
262                 "Excelente": "Oración completa con formato exacto",
263                 "Bueno": "Oración con todos los datos pero formato diferente",
264                 "Regular": "Oración incompleta",
265                 "Insuficiente": "No construye oración"
266             }
267         },
268         {
269             "nombre": "Uso de f-string",
270             "niveles": {
271                 "Excelente": "f-string con todas las variables integradas",

```

```

272         "Bueno": "Usa concatenación correctamente",
273         "Regular": "Mezcla métodos con errores",
274         "Insuficiente": "No integra variables"
275     }
276 }
277 ]
278 },
279 {
280     "numero": 4,
281     "titulo": "Área y Perímetro de Círculo",
282     "objetivo": "Calcular área y perímetro de un círculo dado su radio",
283     "puntaje_maximo": 24,
284     "criterios": [
285         {
286             "nombre": "Importación de math",
287             "niveles": {
288                 "Excelente": "Importa math al inicio del programa",
289                 "Bueno": "Importa math en otra ubicación",
290                 "Regular": "Intenta importar con errores",
291                 "Insuficiente": "No importa librería"
292             }
293         },
294         {
295             "nombre": "Conversión a float",
296             "niveles": {
297                 "Excelente": "Convierte input a float correctamente",
298                 "Bueno": "Convierte pero con sintaxis subóptima",
299                 "Regular": "Convierte a tipo incorrecto",
300                 "Insuficiente": "No convierte"
301             }
302         },
303         {
304             "nombre": "Fórmula del área",
305             "niveles": {
306                 "Excelente": "math.pi * radio**2 correcta",
307                 "Bueno": "Fórmula correcta con notación diferente",
308                 "Regular": "Fórmula con error menor",
309                 "Insuficiente": "Fórmula incorrecta"
310             }
311         },
312         {
313             "nombre": "Fórmula del perímetro",
314             "niveles": {
315                 "Excelente": "2 * math.pi * radio correcta",
316                 "Bueno": "Fórmula correcta con notación diferente",
317                 "Regular": "Fórmula con error menor",
318                 "Insuficiente": "Fórmula incorrecta"
319             }
320         },
321         {
322             "nombre": "Uso de round()",
323             "niveles": {
324                 "Excelente": "Redondea ambos resultados a 2 decimales",
325                 "Bueno": "Redondea solo uno o a diferente precisión",
326                 "Regular": "Intenta redondear con errores",
327                 "Insuficiente": "No redondea"
328             }
329         },
330         {
331             "nombre": "Presentación de resultados",
332             "niveles": {
333                 "Excelente": "Muestra ambos resultados claramente",
334                 "Bueno": "Muestra resultados con formato diferente",
335                 "Regular": "Muestra solo un resultado",
336                 "Insuficiente": "No muestra resultados"
337             }
338         }
339     ]
340 },
341 {
342     "numero": 5,
343     "titulo": "Conversión de Segundos a Horas",
344     "objetivo": "Convertir una cantidad de segundos a horas",
345     "puntaje_maximo": 20,
346     "criterios": [
347         {
348             "nombre": "Solicitud de datos",
349             "niveles": {
350                 "Excelente": "Solicita segundos con mensaje claro",
351                 "Bueno": "Solicita sin mensaje descriptivo",
352                 "Regular": "Solicita dato incorrecto",
353                 "Insuficiente": "No solicita"
354             }
355         },
356         {
357             "nombre": "Conversión a float",
358             "niveles": {
359                 "Excelente": "Convierte input a float correctamente",
360                 "Bueno": "Convierte con sintaxis alternativa",
361                 "Regular": "Convierte a tipo incorrecto",
362                 "Insuficiente": "No convierte"
363             }
364         }
365     ]
366 },

```

```

365         {
366             "nombre": "Fórmula de conversión",
367             "niveles": {
368                 "Excelente": "Divide entre 3600 correctamente",
369                 "Bueno": "Usa factor de conversión equivalente",
370                 "Regular": "Fórmula con error de cálculo",
371                 "Insuficiente": "Fórmula incorrecta"
372             }
373         },
374         {
375             "nombre": "Uso de round()",
376             "niveles": {
377                 "Excelente": "Redondea a 2 decimales",
378                 "Bueno": "Redondea a otra precisión",
379                 "Regular": "Intenta redondear con errores",
380                 "Insuficiente": "No redondea"
381             }
382         },
383         {
384             "nombre": "Presentación",
385             "niveles": {
386                 "Excelente": "Muestra segundos y horas en el resultado",
387                 "Bueno": "Muestra solo las horas",
388                 "Regular": "Formato confuso",
389                 "Insuficiente": "No muestra resultado"
390             }
391         }
392     ]
393 },
394 {
395     "numero": 6,
396     "titulo": "Tabla de Multiplicar",
397     "objetivo": "Mostrar la tabla de multiplicar de un número (del 0 al 9)",
398     "puntaje_maximo": 20,
399     "criterios": [
400         {
401             "nombre": "Solicitud de número",
402             "niveles": {
403                 "Excelente": "Solicita número entero con mensaje claro",
404                 "Bueno": "Solicita sin especificar tipo",
405                 "Regular": "Mensaje confuso",
406                 "Insuficiente": "No solicita"
407             }
408         },
409         {
410             "nombre": "Conversión a int",
411             "niveles": {
412                 "Excelente": "Convierte a int correctamente",
413                 "Bueno": "Convierte a float (funciona pero no óptimo)",
414                 "Regular": "Convierte con errores",
415                 "Insuficiente": "No convierte"
416             }
417         },
418         {
419             "nombre": "Cálculos de multiplicación",
420             "niveles": {
421                 "Excelente": "Todas las multiplicaciones (0-9) correctas",
422                 "Bueno": "Falta alguna multiplicación",
423                 "Regular": "Varias multiplicaciones faltantes o incorrectas",
424                 "Insuficiente": "Cálculos incorrectos"
425             }
426         },
427         {
428             "nombre": "Formato de salida",
429             "niveles": {
430                 "Excelente": "Usa string multilínea, formato claro y ordenado",
431                 "Bueno": "Formato legible pero no óptimo",
432                 "Regular": "Formato desordenado",
433                 "Insuficiente": "Sin formato"
434             }
435         },
436         {
437             "nombre": "Presentación completa",
438             "niveles": {
439                 "Excelente": "Muestra operación y resultado (n x i = resultado)",
440                 "Bueno": "Muestra solo resultados",
441                 "Regular": "Formato incompleto",
442                 "Insuficiente": "No muestra tabla"
443             }
444         }
445     ]
446 },
447 {
448     "numero": 7,
449     "titulo": "Operaciones Aritméticas Básicas",
450     "objetivo": "Realizar suma, resta, multiplicación y división de dos números",
451     "puntaje_maximo": 28,
452     "criterios": [
453         {
454             "nombre": "Solicitud de datos",
455             "niveles": {
456                 "Excelente": "Solicita 2 números con mensajes claros",
457                 "Bueno": "Solicita sin mensajes claros",

```

```

458         "Regular": "Solicita solo 1 número",
459         "Insuficiente": "No solicita"
460     }
461 },
462 {
463     "nombre": "Conversión a float",
464     "niveles": {
465         "Excelente": "Ambas conversiones correctas",
466         "Bueno": "Una conversión correcta",
467         "Regular": "Conversiones con errores",
468         "Insuficiente": "No convierte"
469     }
470 },
471 {
472     "nombre": "Operación suma",
473     "niveles": {
474         "Excelente": "Suma correcta",
475         "Bueno": "Suma con error de presentación",
476         "Regular": "Suma incorrecta",
477         "Insuficiente": "No realiza suma"
478     }
479 },
480 {
481     "nombre": "Operación resta",
482     "niveles": {
483         "Excelente": "Resta correcta",
484         "Bueno": "Resta con error de presentación",
485         "Regular": "Resta incorrecta",
486         "Insuficiente": "No realiza resta"
487     }
488 },
489 {
490     "nombre": "Operación multiplicación",
491     "niveles": {
492         "Excelente": "Multiplicación correcta",
493         "Bueno": "Con error de presentación",
494         "Regular": "Multiplicación incorrecta",
495         "Insuficiente": "No realiza multiplicación"
496     }
497 },
498 {
499     "nombre": "Operación división",
500     "niveles": {
501         "Excelente": "División correcta con round()",
502         "Bueno": "División sin redondeo",
503         "Regular": "División incorrecta",
504         "Insuficiente": "No realiza división"
505     }
506 },
507 {
508     "nombre": "Presentación",
509     "niveles": {
510         "Excelente": "Muestra las 4 operaciones claramente",
511         "Bueno": "Formato aceptable",
512         "Regular": "Formato confuso",
513         "Insuficiente": "No presenta resultados"
514     }
515 }
516 ]
517 },
518 {
519     "numero": 8,
520     "titulo": "Índice de Masa Corporal (IMC)",
521     "objetivo": "Calcular el IMC dado peso y altura",
522     "puntaje_maximo": 24,
523     "criterios": [
524         {
525             "nombre": "Solicitud de peso",
526             "niveles": {
527                 "Excelente": "Solicita peso en kg con mensaje claro",
528                 "Bueno": "Solicita sin especificar unidad",
529                 "Regular": "Mensaje confuso",
530                 "Insuficiente": "No solicita"
531             }
532         },
533         {
534             "nombre": "Solicitud de altura",
535             "niveles": {
536                 "Excelente": "Solicita altura en metros con mensaje claro",
537                 "Bueno": "Solicita sin especificar unidad",
538                 "Regular": "Mensaje confuso",
539                 "Insuficiente": "No solicita"
540             }
541         },
542         {
543             "nombre": "Conversión a float",
544             "niveles": {
545                 "Excelente": "Ambas conversiones correctas",
546                 "Bueno": "Una conversión correcta",
547                 "Regular": "Conversiones con errores",
548                 "Insuficiente": "No convierte"
549             }
550         }

```

```

551         {
552             "nombre": "Fórmula IMC",
553             "niveles": {
554                 "Excelente": "peso / altura**2 correcta",
555                 "Bueno": "Fórmula equivalente correcta",
556                 "Regular": "Fórmula con error menor",
557                 "Insuficiente": "Fórmula incorrecta"
558             }
559         },
560         {
561             "nombre": "Uso de round()",
562             "niveles": {
563                 "Excelente": "Redondea a 2 decimales",
564                 "Bueno": "Redondea a otra precisión",
565                 "Regular": "No redondea",
566                 "Insuficiente": "Error en redondeo"
567             }
568         },
569         {
570             "nombre": "Presentación",
571             "niveles": {
572                 "Excelente": "Muestra IMC con mensaje claro",
573                 "Bueno": "Muestra solo el número",
574                 "Regular": "Formato confuso",
575                 "Insuficiente": "No muestra resultado"
576             }
577         }
578     ]
579 },
580 {
581     "numero": 9,
582     "titulo": "Conversión Celsius a Fahrenheit",
583     "objetivo": "Convertir temperatura de grados Celsius a Fahrenheit",
584     "puntaje_maximo": 20,
585     "criterios": [
586         {
587             "nombre": "Solicitud de temperatura",
588             "niveles": {
589                 "Excelente": "Solicita °C con mensaje claro y símbolo",
590                 "Bueno": "Solicita sin símbolo de grados",
591                 "Regular": "Mensaje confuso",
592                 "Insuficiente": "No solicita"
593             }
594         },
595         {
596             "nombre": "Conversión a float",
597             "niveles": {
598                 "Excelente": "Convierte correctamente",
599                 "Bueno": "Sintaxis alternativa",
600                 "Regular": "Conversión con errores",
601                 "Insuficiente": "No convierte"
602             }
603         },
604         {
605             "nombre": "Fórmula de conversión",
606             "niveles": {
607                 "Excelente": "(9/5) * celsius + 32 correcta",
608                 "Bueno": "Fórmula equivalente (1.8 * celsius + 32)",
609                 "Regular": "Fórmula con error menor",
610                 "Insuficiente": "Fórmula incorrecta"
611             }
612         },
613         {
614             "nombre": "Uso de round()",
615             "niveles": {
616                 "Excelente": "Redondea a 2 decimales",
617                 "Bueno": "Redondea a otra precisión",
618                 "Regular": "No redondea",
619                 "Insuficiente": "Error en redondeo"
620             }
621         },
622         {
623             "nombre": "Presentación",
624             "niveles": {
625                 "Excelente": "Muestra ambas temperaturas con símbolos °C y °F",
626                 "Bueno": "Muestra sin símbolos",
627                 "Regular": "Solo muestra Fahrenheit",
628                 "Insuficiente": "No muestra resultado"
629             }
630         }
631     ]
632 },
633 {
634     "numero": 10,
635     "titulo": "Promedio de Tres Números",
636     "objetivo": "Calcular el promedio de 3 números ingresados",
637     "puntaje_maximo": 24,
638     "criterios": [
639         {
640             "nombre": "Solicitud de datos",
641             "niveles": {
642                 "Excelente": "Solicita 3 números con mensajes claros (primero, segundo, tercero)",
643                 "Bueno": "Solicita sin identificar orden",

```

```

644         "Regular": "Solicita menos de 3 números",
645         "Insuficiente": "No solicita"
646     }
647 },
648 {
649     "nombre": "Conversión a float",
650     "niveles": {
651         "Excelente": "Las 3 conversiones correctas",
652         "Bueno": "2 conversiones correctas",
653         "Regular": "1 conversión correcta",
654         "Insuficiente": "No convierte"
655     }
656 },
657 {
658     "nombre": "Cálculo de suma",
659     "niveles": {
660         "Excelente": "Suma los 3 números correctamente",
661         "Bueno": "Suma con variable intermedia innecesaria",
662         "Regular": "Suma incorrecta",
663         "Insuficiente": "No suma"
664     }
665 },
666 {
667     "nombre": "Cálculo de promedio",
668     "niveles": {
669         "Excelente": "Divide suma entre 3 correctamente",
670         "Bueno": "Fórmula alternativa correcta",
671         "Regular": "Error en división",
672         "Insuficiente": "No calcula promedio"
673     }
674 },
675 {
676     "nombre": "Uso de round()",
677     "niveles": {
678         "Excelente": "Redondea a 2 decimales",
679         "Bueno": "Redondea a otra precisión",
680         "Regular": "No redondea",
681         "Insuficiente": "Error en redondeo"
682     }
683 },
684 {
685     "nombre": "Presentación",
686     "niveles": {
687         "Excelente": "Mensaje claro con el promedio",
688         "Bueno": "Solo muestra número",
689         "Regular": "Formato confuso",
690         "Insuficiente": "No muestra resultado"
691     }
692 }
693 ]
694 }
695 ]
696
697 # Insertar ejercicios y criterios (todos son ejercicios de Python)
698 for ej_data in ejercicios_data:
699     ejercicio = Ejercicio(
700         numero=ej_data["numero"],
701         titulo=ej_data["titulo"],
702         objetivo=ej_data["objetivo"],
703         puntaje_maximo=ej_data["puntaje_maximo"],
704         lenguaje_id=lenguaje_dict["Python"].id
705     )
706     session.add(ejercicio)
707     session.commit()
708
709     for orden, criterio_data in enumerate(ej_data["criterios"], start=1):
710         criterio = Criterio(
711             ejercicio_id=ejercicio.id,
712             nombre=criterio_data["nombre"],
713             orden=orden
714         )
715         session.add(criterio)
716         session.commit()
717
718         for escala_nombre, descripcion in criterio_data["niveles"].items():
719             nivel = NivelDesempeno(
720                 criterio_id=criterio.id,
721                 escala_id=escala_dict[escala_nombre].id,
722                 descripcion=descripcion
723             )
724             session.add(nivel)
725
726     session.commit()
727
728 # Insertar buenas prácticas
729 buenas_practicas = [
730     "Nombres de variables descriptivos y en español",
731     "Comentarios explicativos cuando sea necesario",
732     "Código limpio y legible",
733     "Indentación correcta"
734 ]
735 for bp in buenas_practicas:
736     session.add(BuenaPractica(descripcion=bp))

```



```

737
738 # Insertar errores comunes
739 errores_comunes = [
740     "No convertir tipos de datos cuando es necesario",
741     "Olvidar importar librerías requeridas",
742     "Variables sin inicializar",
743     "Errores de sintaxis en strings (comillas no cerradas)",
744     "División entre cero no manejada (cuando aplique)"
745 ]
746 for ec in errores_comunes:
747     session.add(ErrorComun(descripcion=ec))
748
749 session.commit()
750
751
752 def main():
753     """Función principal para crear la base de datos y cargar los datos"""
754     print("Configurando conexión a PostgreSQL...")
755     print(f"URL de conexión: {DATABASE_URL}")
756     print("\nNOTA: Modifica DATABASE_URL con tus credenciales antes de ejecutar.")
757     print("Ejemplo: postgresql://usuario:contraseña@localhost:5432/rubricas_secuenciales\n")
758
759     try:
760         engine = get_engine()
761         print("Creando tablas...")
762         create_tables(engine)
763         print("Tablas creadas exitosamente.")
764
765         session = get_session(engine)
766         print("Insertando datos iniciales...")
767         insertar_datos_iniciales(session)
768         print("Datos insertados exitosamente.")
769
770         session.close()
771         print("\nBase de datos configurada correctamente.")
772
773     except Exception as e:
774         print(f"Error: {e}")
775         print("\nAsegúrate de:")
776         print("1. Tener PostgreSQL instalado y ejecutándose")
777         print("2. Crear la base de datos 'rubricas_secuenciales'")
778         print("3. Configurar las credenciales correctas en DATABASE_URL")
779
780
781 if __name__ == "__main__":
782     main()
783

```