Bind runtime args

Sometimes we want to invoke a Runnable within a Runnable sequence with constant arguments that are not part of the output of the preceding Runnable in the sequence, and which are not part of the user input. We can use `Runnable.bind()` to easily pass these arguments in.

Suppose we have a simple prompt + model sequence:

```
%pip install --upgrade --quiet  langchain
langchain-openai
```

```
from langchain.schema import StrOutputParser
from langchain_core.prompts import
ChatPromptTemplate
from langchain_core.runnables import
RunnablePassthrough
from langchain_openai import ChatOpenAI
```

```
prompt = ChatPromptTemplate.from_messages(
    [
```

```
        (
            "system",
            "Write out the following equation
using algebraic symbols then solve it. Use
the
format\n\nEQUATION:...\nSOLUTION:...\n\n",
        ),
        ("human", "{equation_statement}"),
    ]
)
model = ChatOpenAI(temperature=0)
runnable = (
    {"equation_statement":
RunnablePassthrough()} | prompt | model |
StrOutputParser()
)

print(runnable.invoke("x raised to the third
plus seven equals 12"))
```

```
EQUATION: x^3 + 7 = 12

SOLUTION:
Subtracting 7 from both sides of the
equation, we get:
x^3 = 12 - 7
x^3 = 5

Taking the cube root of both sides, we get:
x = ∛5
```

```
Therefore, the solution to the equation x^3 +
7 = 12 is x = ∛5.
```

and want to call the model with certain `stop` words:

```
runnable = (
    {"equation_statement":
RunnablePassthrough()}
    | prompt
    | model.bind(stop="SOLUTION")
    | StrOutputParser()
)
print(runnable.invoke("x raised to the third
plus seven equals 12"))
```

```
EQUATION: x^3 + 7 = 12
```

Attaching OpenAI functions

One particularly useful application of binding is to attach OpenAI functions to a compatible OpenAI model:

```
function = {
    "name": "solver",
    "description": "Formulates and solves an
equation",
    "parameters": {
        "type": "object",
        "properties": {
            "equation": {
                "type": "string",
                "description": "The algebraic
expression of the equation",
            },
            "solution": {
                "type": "string",
                "description": "The solution
to the equation",
            },
        },
        "required": ["equation", "solution"],
    },
}
```

```
# Need gpt-4 to solve this one correctly
prompt = ChatPromptTemplate.from_messages(
    [
        (
            "system",
            "Write out the following equation
using algebraic symbols then solve it.",
```

```
        ),
        ("human", "{equation_statement}"),
    ]
)
model = ChatOpenAI(model="gpt-4",
temperature=0).bind(
    function_call={"name": "solver"},
functions=[function]
)
runnable = {"equation_statement":
RunnablePassthrough()} | prompt | model
runnable.invoke("x raised to the third plus
seven equals 12")
```

```
AIMessage(content='', additional_kwargs=
{'function_call': {'name': 'solver',
'arguments': '{\n"equation": "x^3 + 7 =
12",\n"solution": "x = ∛5"\n}'}},
example=False)
```

Attaching OpenAI tools

```
tools = [
    {
        "type": "function",
        "function": {
            "name": "get_current_weather",
```

```
            "description": "Get the current
weather in a given location",
            "parameters": {
                "type": "object",
                "properties": {
                    "location": {
                        "type": "string",
                        "description": "The
city and state, e.g. San Francisco, CA",
                    },
                    "unit": {"type":
"string", "enum": ["celsius", "fahrenheit"]},
                },
                "required": ["location"],
            },
        },
    }
]
```

```
model = ChatOpenAI(model="gpt-3.5-turbo-
1106").bind(tools=tools)
model.invoke("What's the weather in SF, NYC
and LA?")
```

```
AIMessage(content='', additional_kwargs=
{'tool_calls': [{'id':
'call_zHN0ZHwrxM7nZDdqTp6dkPko', 'function':
{'arguments': '{"location": "San Francisco,
```

```
CA", "unit": "celsius"}', 'name':
'get_current_weather'}, 'type': 'function'},
{'id': 'call_aqdMm9HBSlFW9c9rqxTa7eQv',
'function': {'arguments': '{"location": "New
York, NY", "unit": "celsius"}', 'name':
'get_current_weather'}, 'type': 'function'},
{'id': 'call_cx8E567zcLzYV2WSWVgO63f1',
'function': {'arguments': '{"location": "Los
Angeles, CA", "unit": "celsius"}', 'name':
'get_current_weather'}, 'type':
'function'}]})
```