```
from PIL import Image
from transformers import AutoModelForCausalLM, AutoTokenizer

from swarm_models.base_multimodal_model import BaseMultiModalModel


class MoonDream(BaseMultiModalModel):
    """
     MoonDream is a multi-modal model that combines text and image inputs to generate descriptive answers for images.

    Args:
        model_name (str): The name or path of the pre-trained model to be used.
        revision (str): The specific revision of the pre-trained model to be used.

    Attributes:
        model_name (str): The name or path of the pre-trained model.
        revision (str): The specific revision of the pre-trained model.
        model (AutoModelForCausalLM): The pre-trained model for generating answers.
        tokenizer (AutoTokenizer): The tokenizer for processing text inputs.

    """

    def __init__(
        self,
        model_name: str = "vikhyatk/moondream2",
```

```
    revision: str = "2024-03-04",
    system_prompt: str = None,
    *args,
    **kwargs,
):
    super().__init__()
    self.model_name = model_name
    self.revision = revision
    self.system_prompt = system_prompt

    self.model = AutoModelForCausalLM.from_pretrained(
        model_name,
        trust_remote_code=True,
        revision=revision,
        *args,
        **kwargs,
    )
    self.tokenizer = AutoTokenizer.from_pretrained(
        model_name, revision=revision
    )

def run(self, task: str, img: str):
    """
    Runs the MoonDream model to generate a descriptive answer for the given image.

    Args:
```

```
    task (str): The task or question related to the image.
    img (str): The path or URL of the image file.

Returns:
    str: The descriptive answer generated by the MoonDream model.

"""
image = Image.open(img)
enc_image = self.model.encode_image(image)
return self.model.answer_question(
    enc_image, f"{self.system_propmpt} {task}", self.tokenizer
)
```