To integrate smart devices and set up data collection using IBM Cloud Functions, we will need to create a serverless function that can receive data from your devices and store it in IBM Cloud Object Storage. Here's a high-level guide along with a Python program:

**Step 1: Device Integration**

1. Select Smart Devices :

Choose the IoT devices you want to integrate into your smart home environment. Ensure these devices are capable of sending data, and you have access to their APIs or data streams.

1. Device Setup :

Set up your smart devices according to their manufacturer's instructions. Connect them to your local network and make sure they can communicate with the cloud.

1. **IoT Platform :**

If your devices use specific IoT platforms, set up instances of these platforms. For this example, we'll use IBM Cloud Functions.

**Step 2: Set Up IBM Cloud Functions**

1. **Create an IBM Cloud Functions Action :**

Log in to your IBM Cloud account and navigate to the IBM Cloud Functions service. Create a new action that will collect and process data from your IoT devices.

1. **Choose a Programming Language :**

Select a programming language supported by IBM Cloud Functions. We'll use Python for this example.

1. **Write Data Collection Code :**

Write a Python script for your IBM Cloud Function to collect data from IoT devices and store it in IBM Cloud Object Storage.

**Python Code :**

import requests

import ibm\_boto3

from ibm\_botocore.client import Config

def main(params):

device\_api\_endpoint = " https://config.cloud-object-storage.cloud.ibm.com "

try:

response = requests.get(device\_api\_endpoint)

if response.status\_code == 200:

data = response.json()

bucket\_name = "crn:v1:bluemix:public:cloud-object-storage:global:a/9e95d6860c864c09998b14bd52f93b68:b00228fc-6641-4a3c-92ea-e4e78cb4364e:bucket:ibmnaanmudhalvan"

object\_name = "iot\_data.json"

cos\_credentials = {

"apikey": " vPc\_jSjpKkr4ENC-y8iFlg0jwtQcVfdKFvCQmsu8SXSF",

"iam\_service\_endpoint": " crn:v1:bluemix:public:iam-identity::a/9e95d6860c864c09998b14bd52f93b68::serviceid:ServiceId-fe940556-dddb-499a-b494-af49781d1da7",

"resource\_instance\_id": " crn:v1:bluemix:public:cloud-object-storage:global:a/9e95d6860c864c09998b14bd52f93b68:b00228fc-6641-4a3c-92ea-e4e78cb4364e::",

"endpoint": " https://control.cloud-object-storage.cloud.ibm.com/v2/endpoints",

}

cos\_client = ibm\_boto3.client(

"s3",

ibm\_api\_key\_id=cos\_credentials["apikey"],

ibm\_service\_instance\_id=cos\_credentials["resource\_instance\_id"],

ibm\_auth\_endpoint=cos\_credentials["iam\_service\_endpoint"],

config=Config(signature\_version="oauth"),

endpoint\_url=cos\_credentials["endpoint"],

)

cos\_client.put\_object(Bucket=bucket\_name, Key=object\_name, Body=str(data))

return {"message": "Data collected and stored successfully."}

else:

return {"error": "Failed to collect data from the IoT device."}

except Exception as e:

return {"error": str(e)}

Make sure to replace the placeholders with your actual device API endpoint, Object Storage bucket name, and IBM Cloud Object Storage credentials.

**Step 3: Test Data Collection**

Test your IBM Cloud Function by invoking it and checking if the data is collected and stored in IBM Cloud Object Storage.

By following these steps and using the sample Python program, you can integrate smart devices into your smart home environment and set up data collection with IBM Cloud Functions. The collected data can be used for further processing, automation, and analysis.