

```

# Copyright (c) 2022-2023 Robert Bosch GmbH and Microsoft Corporation
#
# This program and the accompanying materials are made available under the
# terms of the Apache License, Version 2.0 which is available at
# https://www.apache.org/licenses/LICENSE-2.0.
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
# WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
# License for the specific language governing permissions and limitations
# under the License.
#
# SPDX-License-Identifier: Apache-2.0

"""A sample Velocitas vehicle app for collecting vehicle data."""

import logging

from vehicle import Vehicle # type: ignore

from sdv.util.log import ( # type: ignore
    get_opentelemetry_log_factory,
    get_opentelemetry_log_format,
)
from sdv.vdb.reply import DataPointReply
from sdv.vehicle_app import VehicleApp

logging.setLogRecordFactory(get_opentelemetry_log_factory())
logging.basicConfig(format=get_opentelemetry_log_format())
logging.getLogger().setLevel("DEBUG")
logger = logging.getLogger(__name__)

class DiagnosticsApp(VehicleApp):
    """
    Sample Velocitas Vehicle App.

    """

    def __init__(self, vehicle_client: Vehicle):
        super().__init__()
        self.Vehicle = vehicle_client

    async def on_start(self):
        """Run when the vehicle app starts"""
        await self.Vehicle.Speed.subscribe(self.on_speed_change)
        await self.Vehicle.IsMoving.subscribe(self.moving)
        await self.Vehicle.Chassis.Accelerator.PedalPosition.subscribe(self.on_accelerator_pedal_pos_change)
        await self.Vehicle.Body.Windshield.Front.Wiping.System.Mode.subscribe(self.on_windshield_mode_change)

    async def on_speed_change(self, data: DataPointReply):
        vehicle_speed = data.get(self.Vehicle.Speed).value
        print("Current Speed is: ", vehicle_speed)

    async def on_accelerator_pedal_pos_change(self, data: DataPointReply):
        acc_pedal_pos = data.get(self.Vehicle.Chassis.Accelerator.PedalPosition).value
        print("Accelerator Pedal Position is:", acc_pedal_pos)

    async def moving(self, data: DataPointReply):
        isMoving = data.get(self.Vehicle.IsMoving).value
        print("Moving: ", isMoving)

    async def on_windshield_mode_change(self, data: DataPointReply):
        windshield_mode = data.get(self.Vehicle.Body.Windshield.Front.Wiping.System.Mode).value
        print("WindShield Mode: ", windshield_mode)

```