

Test af CPR Broker-tilføjelser

Der er tilføjet 5 nye metoder til CPR Brokerens API: ReadSnapshot, ReadPeriod, ListSnapshot, ListPeriod og SubscribeOnCriteria.

Disse kan tilgås, på samme måde, som de allerede eksisterende metoder. De initiale indstillinger (autentificering, nøgleudveksling ol.) er derfor de samme, som I anvender i jeres nuværende tilgang til CPR Broker.

Testserverens IP er: 10.1.2.197

Nedenfor følger eksempler på hvordan disse metoder kan tilgås.

ReadSnapshot

Denne metode tager et LaesOejbliksbilledeInputType-objekt og returnerer et LaesOutType-objekt.

Her er et kodeeksempel, som kan anvendes til testing af metoden:

```
var uuid = PartService.GetUuid([cpr-nummer]);
Part.LaesOejbliksbilledeInputType input = new Part.LaesOejbliksbilledeInputType() {
    UUID = uuid.UUID,
};
LaesOutputType person = PartService.ReadSnapshot(input);
```

ReadPeriod

Denne metode tager et LaesPeriodeInputType-objekt og returnerer et LaesOutType-objekt.

```
var uuid = PartService.GetUuid([cpr-nummer]);
Part.LaesPeriodInputType test = new Part.LaesPeriodInputType() {
    UUID = uuid.UUID,
    VirkningFraDato = [fradato],
    VirkningTilDato = [tildato],
};
LaesOutputType person = PartService.ReadSnapshot(input);
```

ListSnapshot

Denne metode tager et ListOejbliksbilledeInputType-objekt og returnerer et ListOutType-objekt.

```
Part.ListOejbliksbilledeInputType listInput = new Part.ListOejbliksbilledeInputType() {
    UUID = [cpr1, cpr2, cpr3, cpr4, cpr5, cpr6],
    VirkningDato = [dato],
};
```

ListPeriod

Denne metode tager et ListPeriodeInputType-objekt og returnerer et ListOutType-objekt.

```
Part.ListPeriodInputType periodInput = new ListPeriodInputType() {
    UUID = [cpr1, cpr2, cpr3, cpr4, cpr5, cpr6],
    VirkningFraDato = [fradato],
    VirkningTilDato = [tildato],
};
```

SubscribeOnCriteria

Denne metode tager et SoegObjektType- samt et ChannelBaseType-objekt og returnerer et ChangeSubscriptionType-objekt.

```
var SoegObjekt = new Subscriptions.SoegObjektType() {
    UUID = null,
    BrugervendtNoegleTekst = null,
    SoegAttributListe = new Subscriptions.SoegAttributListeType() {
        SoegEgenskab = new Subscriptions.SoegEgenskabType[] {
            new Subscriptions.SoegEgenskabType() {
                AndreAdresser = new Subscriptions.AdresseType() {
                    Item = new Subscriptions.DanskAdresseType() {
                        AddressComplete = new Subscriptions.AddressCompleteType() {
                            AddressAccess = new Subscriptions.AddressAccessType() {
                                MunicipalityCode = "104"
                            }
                        }
                    }
                }
            }
        }
    }
};
var fileShareChannel = new Subscriptions.FileShareChannelType() { Path="[sti_til_fil]" };
var subscription = SubscriptionsService.SubscribeOnCriteria(fileShareChannel, SoegObjekt);
var subscriptionID = subscription.SubscriptionId;
```