Auction Suffix

Arguments: suffix: SettlementSuffixData

```
type AuctionPoint = {
  delay: number; // point in time of this point relatively to previous point
  coefficient: number; // coefficient rate bump from the end of an auction
};

type AuctionWhitelistItem = {
  address: string;
  allowance: number; // seconds
};

type SettlementSuffixData = {
  points: AuctionPoint[]; // represents auction points with rates and delays
  whitelist: AuctionWhitelistItem[]; // combination of the resolver address and allowance in seconds, which represents when the resolver can start full fill the order
  publicResolvingDeadline?: number; // represents time in seconds when the order started to be public (can be filled by any one)
  takerFeeReceiver?: string; // address of the fee receiver
  takerFeeRatio?: string; // taker ratio, 10000000 = 1%
};
```

Example:

```
import { AuctionSuffix } from "@1inch/fusion-sdk";

const suffix = new AuctionSuffix({
  points: [
    {
      coefficient: 20000,
      delay: 12,
    },
  ],
  whitelist: [
    {
      address: "0x00000000219ab540356cbb839cbe05303d7705fa",
      allowance: 0,
    },
  ],
});
```

```
suffix.build();
// #=> '000c004e2000000000000000000219ab540356cbb839cbe05303d7705faf486570009'
```

static AuctionSuffix.decode

Arguments: interactions: string

Example:

```
import { AuctionSuffix } from "@1inch/fusion-sdk";

const encodedSuffix =
  "000c004e2000000000000000000219ab540356cbb839cbe05303d7705fa63c0566a09";

const suffix = AuctionSuffix.decode(encodedSuffix);

suffix.build();
// #=> '000c004e2000000000000000000219ab540356cbb839cbe05303d7705fa63c0566a09'
```

Previous
‹ Auction Salt

Next
WebSocket API ›